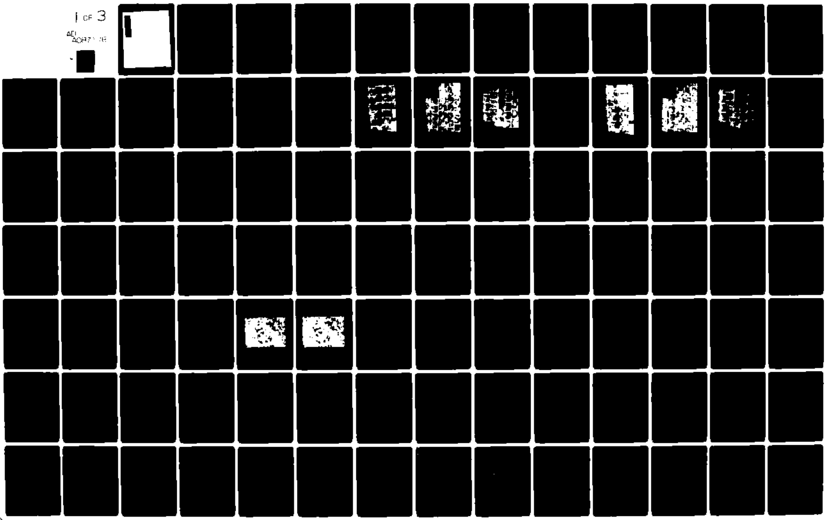
AD-A087 518 IBM FEDERAL SYSTEMS DIV GAITHERSBURG MD F/G 20/6
INTERACTIVE DIGITAL IMAGE PROCESSING INVESTIGATION. PHASE II.(U)
APR 80 W C RICE, J S SHIPMAN, R J SPIELER DAAK70-77-C-0166
UNCLASSIFIED ETL-0221 NL

1 OF 3



INTERACTIVE DIGITAL IMAGE PROCESSING INVESTIGATION, PHASE II

LEVEL (12)

ETL - 0221

IBM

Destroy this report when no longer needed.
Do not return it to the originator.

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

The citation in this report of trade names of commercially available products does not constitute official endorsement or approval of the use of such products.

18 ETL - 0221 19

6 INTERACTIVE DIGITAL IMAGE PROCESSING INVESTIGATION. PHASE II.

10 W. C. Rice
J. S. Shipman
R. J. Spieler

International Business Machines Corporation
Federal System Division
18100 Frederick Pike
Gaithersburg, Maryland 20760

11 Apr[redacted] [redacted]80   12 194

9 Final Rep[redacted]t.

C

Prepared For

U. S. Army Engineer Topographic Laboratories
Fort Belvoir, Virginia 22060

15 DAAK70-77-C-0166

Approved For Public Release; Distribution Unlimited

174 750

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER ETL-0221	2. GOVT ACCESSION NO. AD-A087 518	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Interactive Digital Image Processing Investigation, Phase II A076 342		5. TYPE OF REPORT & PERIOD COVERED Contract Report 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) W. C. Rice J. S. Shipman R. J. Spieler		8. CONTRACT OR GRANT NUMBER(s) DAAK70-77-C-0166
9. PERFORMING ORGANIZATION NAME AND ADDRESS International Business Machines Corporation Federal Systems Division 18100 Frederick Pike Gaithersburg, Maryland 20760		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS U. S. Army Engineer Topographic Laboratories Fort Belvoir, Virginia 22060		12. REPORT DATE April 1980 13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Remote sensing, pattern recognition, feature extraction, digital image processing, photogrammetry, classification

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The objective of the second phase of this investigation was to continue the development of the interactive multi-channel image classification capabilities of the DIAL system. This development proceeded in four directions. Formal demonstrations and a "hands on" course in the DIAL algorithms implemented

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered) under the first phase of the investigation were given. Additional DIAL algorithms to support classification were developed, coded, and tested. These included a Program Module (PM) to apply the Karhunen - Loève transformation to a multi-channel image, which has the effect of reducing the dimensionality of an image without significantly decreasing its information content. In addition two algorithms in refining class assignment by relaxation methods were developed. One was selected, then coded on DIAL and was applied to a classification of a LACIE intensive site, where it removed "speckle", sharpened field boundaries, and increased the overall classification accuracy. A task to program the computationally intensive part of the maximum likelihood method on the STARAN was undertaken jointly with ETL. Finally an experiment in the maximum likelihood classification of a LANDSAT scene using the DIAL PMs was performed in cooperation with an ETL botanist. This experiment demonstrated the utility of the interactive classification algorithms in the study of the relationship between flora and geological structures.

PREFACE

This is the final report for the second phase of the Interactive Digital Image Processing Investigation, completed for the U. S. Army Engineer Topographic Laboratories at Fort Belvoir, Virginia. The purpose of the investigation was to conduct a course in the use of the classification algorithms for multi-channel images implemented on the DIAL system under the first phase of the contract; to implement additional algorithms including classification refinement by relaxation methods; and to conduct an experiment in the applicability of the algorithms to the classification of the flora in a Landsat scene of the Fort Bliss, TX area.

The Contracting Officer's Representative responsible for the above investigation was Mr. Robert Rand.

CONTENTS

Section		Page

Section		Page

LIST OF ILLUSTRATIONS

LIST OF TABLES

Section 1

INTRODUCTION

1.1 BACKGROUND

In the first phase of this investigation two feature extraction (classification) algorithms were implemented as DIAL callable program modules (PMs). Both algorithms are intended to be applied to multi-channel images, that is, images in which a vector of values is associated with each image pixel. One algorithm chosen is a maximum likelihood classification method, an example of supervised classification which has been implemented as the DIAL PM MAXLIK; the other algorithm chosen is a clustering method based on the ISODATA concept, an example of unsupervised classification which has been implemented as the DIAL PM CLUSTER. These PMs required the implementation of a PM called INTERL to produce the multi-channel (composite) images which are input data for MAXLIK and CLUSTER; a PM called FIELDEF to define interactively fields and classes in the image to be classified; and a PM called CLASTAT to compute class signatures. These PMs are described in Reference 1.

The second phase of this investigation was concerned with applying the classification modules to LANDSAT scenes of interest; with conducting classes in the theory and operation of the classification modules; and with further enhancing the DIAL classification capability by developing PMs for ratioing images, for reducing the dimensionality of a multi-channel image without significantly decreasing the information content through the Karhunen - Loève transformation, and for refining the class assignment of a classified image by relaxation methods. Further, the computationally intensive part of the MAXLIK module was programmed for the STARAN computer in a shared programming effort with ETL.

1.2 REPORT ORGANIZATION

Section 1 contains the background of the present effort and a description of the organization and content of the report.

Section 2 describes the Ft. Bliss experiment in which the classification modules were applied to LANDSAT images of a scene near Ft. Bliss, TX. The additional software developed for this application is described in the context of its application to the Ft. Bliss imagery.

Section 3 discusses the Karhumen - Loève transformation, a unitary principal components transformation which effectively reduces the dimensionality of an image without significantly decreasing the information content. The transformed images can be classified in significantly less computer time than the original image with very little loss in classification accuracy.

Section 4 is concerned with relaxation methods for refining the classification of a scene. This is a special case of the general idea of scene labeling by relaxation which has been under development recently. The modules developed, which permit a range of algorithms within the same processing framework, are described and their use illustrated.

Section 5 describes the programming of the class distance and class assignment computations of the MAXLIK PM on the STARAN computer. The 6400/STARAN interfaces are defined, and the data manipulation in the

6400 for handling data shipped to and received from the STARAN are detailed.

Section 6 gives the content of the classification courses conducted at ETL.

Section 7 documents the software developed under the present contract. The general format for each program classification is: user information, control flow, and program description.

Section 8 contains recommendations for further work based on the results of the present study.

Section 2

FORT BLISS EXPERIMENT

2.1 EXPERIMENT BACKGROUND AND OBJECTIVES

M.B. Satterwhite and colleagues at ETL have determined the relationships between landforms and plant communities for a 650,000 hectare area in the northern Chihuahuan Desert in north-central New Mexico and western Texas (Ref. 9). The principal technique used was phytosociological analysis of 1:50,000 scale stereoscopically viewed panchromatic aerial photography. The resulting landform image is shown in Figures 2.1-1, 2.1-2, and 2.1-3; the legends for the landforms represented as well as their respective percentages of the study area are given in Table 2.1-1.

The resulting land cover map is shown in Figure 2.1-4 - 2.1-6; the legends for the land cover types represented as well as their respective percentages of the study area are given in Table 2.1-2. Four major landform - soil condition groups were identified for which the specific plant community occurred more than 30% of its overall distribution in the study area.

In view of the belief that LANDSAT data can be used for land cover classification to Anderson's Level II (Ref. 10), and in certain cases with sufficient effort to Level III, there is interest in continuing the landform/land cover analysis using LANDSAT data plus other sources, in particular digital elevation data, which would be added to the LANDSAT data as an additional channel. As a long-range objective this analysis

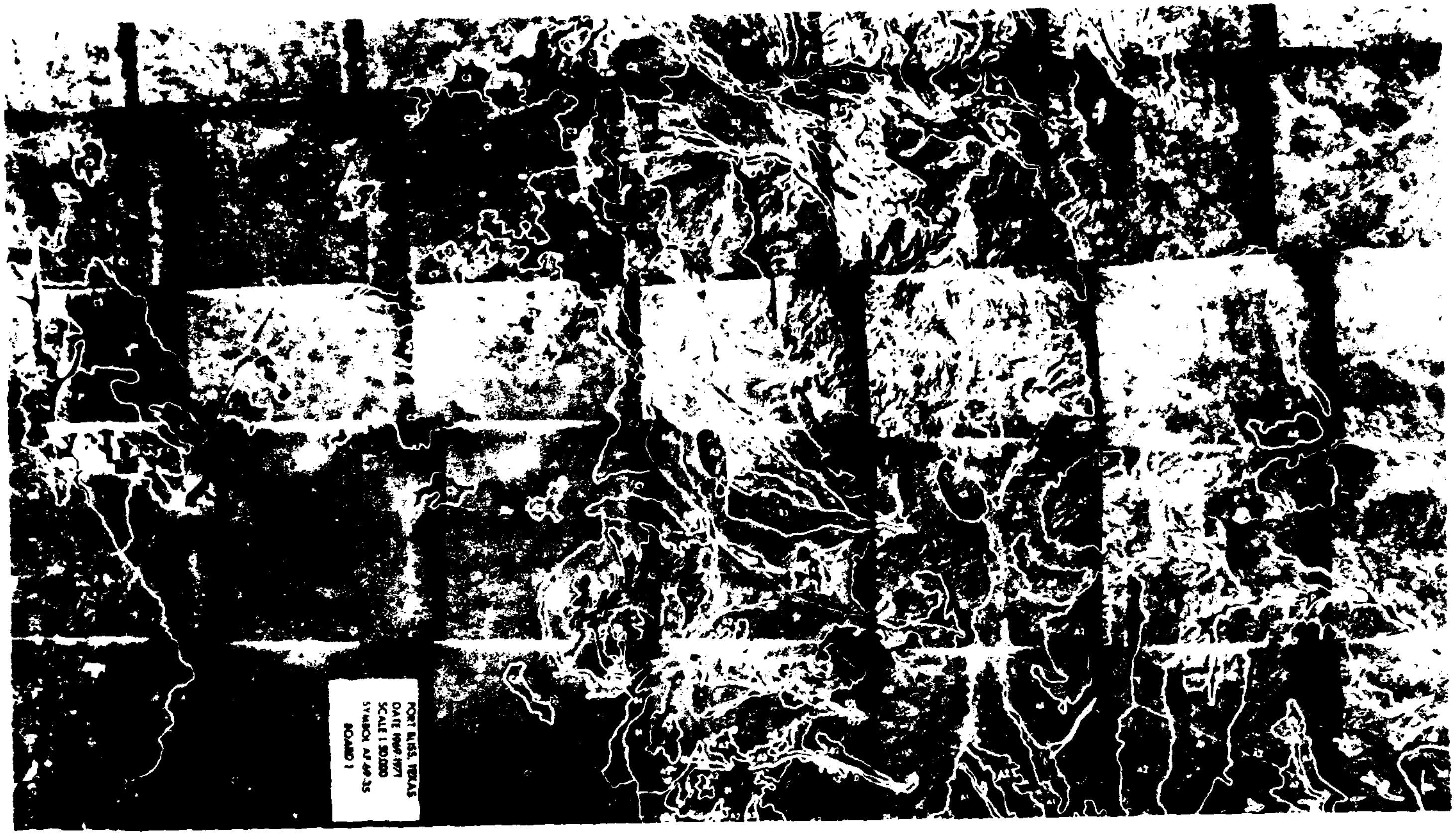


Figure 2.1-1. Landform Map of the Study Area (Photo Mosiac 1). Scale 1:270,000

Figure 2.1-2. Landform Map of the Study Area (Photo Mosaic 2). Scale 1:270,000

Figure 2.1-3. Landform Map of the Study Area (Photo Mosaic 3). Scale 1:270,000

Table 2.1-1. Landform Units and Their Percentages in the Study Area

Major Landform Unit	Map Symbol	Landform Subunit	Percentage of Study Area
Mountains/Hills	A1	Mesa	15.8
	A2	Highly dissected hills	16.1
	A3	Rugged, sharp-crested mountains	2.3
Alluvial Fans	B1	Primary high elevation fans	10.8
	B2	Secondary high elevation fans	1.2
	B3	Mottled, intermediate elevation fans	6.8
	B4	Dark-toned, lowest elevation fans	3.6
	B5	Fans covered with deep aeolian sand	3.5
	B6	High elevation anomalous fans	0.7
Basin Areas	C1	Light-toned, speckled sand dunes	30.2
	C2	Dark-toned, rough-textured dunes	2.6
	C3	Low, smooth areas	2.5
	C4	Small, dark-toned depressions	1.5
Washes	D	---	2.0
			99.6

Figure 2.1-4. Land Cover Map of the Study Area (Photo Mosaic 1).
Scale 1:270,000

Figure 2.1-5. Land Cover Map of the Study Area (Photo Mosaic 2). Scale 1:270,000

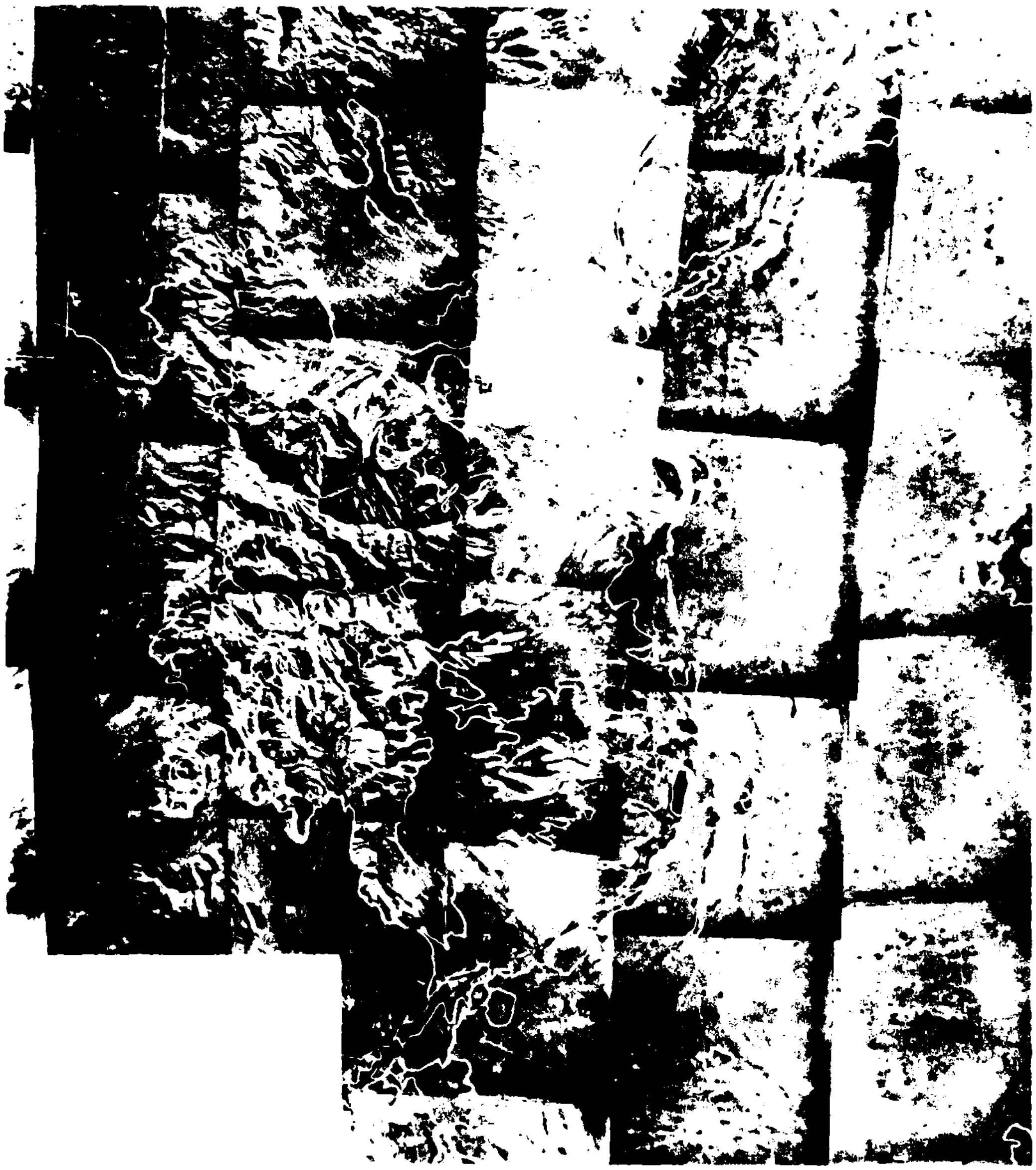


Figure 2.1-6. Land Cover Map of the Study Area (Photo Mosaic 3). Scale 1:270,000

Table 2.1-2. Land Cover Mapping Units and Their Percentages of the Study Area

Physiognomic Group	Group Percent of Area	Land Cover Unit	Percent of Area
Grassland	37.4		
		Grassland (10)	21.1
		Grass-*Larrea tridentata* (11)	0.7
		Grass-*Flourensia cernua* (12)	1.3
		Grass-*Acacia constricta* (13)	<0.1
		Grass-*Artemisia filifolia* (14)	0.7
		Grass-*Prosopis glandulosa* (15)	2.2
		Grass-*Parthenium incanum* (16)	11.3
Shrubland	58.2		
		Larrea tridentata (20)	9.6
		Larrea tridentata-Grass (21)	4.6
		Larrea tridentata-Grass-*Parthenium incanum* (22)	0.6
		Larrea tridentata-*Prosopis glandulosa*-Grass (23)	4.1
		Larrea tridentata-*Fourensia cernua*-Grass (25)	2.4
		Acacia constricta-Grass (30)	1.2
		Acacia constricta-*Laurrea tridentata*-Grass (31)	0.8
		Flourensia cernua-Grass (40)	<0.1
		Flourensia cernua-*Larrea tridentata* (41)	2.2
		Prosopis glandulosa-*Atriplex canescens*- *Xanthocephalum Sarothrae* (50)	27.5
		Prosopis glandulosa-*Larrea tridentata*-Grass (51)	0.9
		Prosopis glandulosa-*Artemisia filifolia*-Grass (52)	0.7
		Artemisia filifolia-Grass (60)	2.8
		Artemisia filifolia-*Prosopis glandulosa*-Grass (61)	0.7

(continued on next page)

Table 2.1-2. Land Cover Mapping Units and Their Percentages of the Study Area (Cont.)

Physiognomic Group	Group Percent of Area	Land Cover Unit	Percent of Area
Forestland	2.2	Juniperus monosperma-Quercus undulata (70)	2.2
Other	2.3	Bare ground (90)	0.2
		Water bodies (91)	0.1
		Urban and built-up areas (92)	2.0
Total	100.1		100.1

would develop models which would apply not only to the desert, but to other vegetation/crop cover regions and would consider phenological climate/species effects on the spectral signatures.

As the first step in this extended analysis an experiment in the classification of LANDSAT images of the Fort Bliss, TX area was carried out in a joint effort with M. B. Satterwhite. The experiment made use of the set of classification PMs developed under the first phase of the contract. In addition, PM MSSRFT was developed to deinterleave LANDSAT computer compatible tapes (CCTs), and PM RATIOF was developed to form the ratio of two single-channel images. These PMs are described in Sections 7.2 and 7.1 respectively.

The experiment had as its primary objective the testing of the feasibility of the automatic classification of LANDSAT imagery in the Level III classification of land cover. The LANDSAT images chosen covered an area of New Mexico and Texas which included the study area of Reference 9. A secondary objective of the experiment was to determine the utility of ratioed and differenced LANDSAT images in the visual analysis of land cover and landforms.

2.2 EXPERIMENT PREPARATION

The images chosen for the experiment were LANDSAT scenes 2059-16590 of March 22, 1975 and 2221-16575 of August 31, 1975. These were deinterleaved using MSSRFT and stored. Subimages containing about one-quarter of the original scene and surrounding the study area were created using the DIAL PM GENSUBN. The subscenes were stored on PK0012, which was the principal working pack for the experiment, as images S1AB4S - S1AB7S for the August 31 scene and S2AB4S - S2AB7S for the March 22 scene (the suffix "S" indicates subimage). In addition four-channel composite images were created using INTERL and stored on PK0012 as S1AB and S2AB.

Since it was anticipated that both the August and March scenes would be classified in order to determine whether seasonal effects could be detected, it was necessary to register the two images. Since a full temporal registration was not practical, it was assumed that S2AB4 was translated with respect to S1AB4. The amount of translation was determined by identifying the same features in both images, located near the corners of the images, and determining their coordinates using the FIELDEF PM. Subimages S2AB4RS - S2AB7RS were created which were translates of S2AB4S - S2AB7S by the amount found (42 pixels in the line coordinate and 111 pixels in the sample coordinate), and which therefore are approximately registered to S1AB4S - S1AB7S respectively.

In preparation for the evaluation of the usefulness of ratioed images, a let of images which were ratios of pairs of bands of the August 31 subscene were created. The combinations were 4:5, 4:6, 4:7, 5:4, 5:6, 5:7 and 7:4; these were chosen because they have been found useful by other researchers. The ratioed images where also stored on pack PK0012. The naming scheme was to call, for example, the ratio of band 4 of the August subscene to band 5 of the August subscene RAT45S1.

2.3 CONDUCT OF THE EXPERIMENT

2.3.1 Field and Class Definition

Field definition is typically a highly interactive and iterative process. Training fields are identified by visually comparing a displayed digital image with ground truth, in this case the landform and land cover maps, Figures 2.1-1 through 2.1-6. The fields defined using the FIELDEF PM are stored (in the form of image coordinates of the vertices) in a field/class file. Generally several fields of each class of interest are defined. Classes are generated from the training fields in the CLASTAT PM, which computes the mean vectors and covariance matrices of the pixels contained in the respective fields. The usual practice is to generate a class for each training field.

Classes corresponding to training fields believed to contain the same material are then compared with respect to their mean vectors, and for those classes whose degree of distinguishability is in doubt, Bhattacharyya distances are computed. As a working rule a Bhattacharyya distance of 0.3 was taken as the value which separated similiar classes (>0.3) from dissimiliar classes (<0.3).

In those cases in which separate training fields lead to very similiar classes it is generally a matter of indifference whether the similiar classes are combined into a single class or one of the classes is retained for the maximum likelihood classification. In other cases in which two separate training fields thought to contain the same material lead to distinctly dissimiliar areas, both classes were retained for the maximum likelihood classification in the hope that the results of the classification process will clarify the situation.

Since classification of the area defined by the composite images S1AB and S2AB was impractical (each contains 2,062,500 pixels, and a maximum likelihood classification into ten classes would take almost three hours on the CDC 6400) three subareas in the Meyer, Dona Ana, and Orogrande regions were outlined and the training fields defined were confined to one of the three subregions.

In the first attempt at defining a set of training fields for a sub-region, the fields were designated by (more or less) pronounceable names suggested by the material which the field was believed to contain. Although the set was used in a preliminary classification, the field names proved to be so unwieldy and inconvenient for referencing and discussion that this field/class file was abandoned and a new field/class file initiated which used as its naming convention the type numbers of Table 2.1-2. With the new naming convention, it was possible to interactively build up the field/class file AUG2 and complete the classification of both the Aug 31 and March 22 scenes. While ordinarily the naming of objects would not be expected to make any difference in a computation, the experience described indicates the importance of convenient naming conventions in an interactive system that must also be exercised iteratively.

With a selection of fields named by the new naming convention, a field-file was built from which classes were computed. A set of ten classes was selected which were represented in the Meyer area, and that sub-region classified using MAXLIK. In order to gather statistics on the adequacy of the set of classes, a set of fields which included all of the training fields from which the classes were derived was specified. In general the classes specified in the classification of a given region should include all of classes of material likely to be found in that region. Here the present limit of ten classes in MAXLIK imposes a constraint on the selection. This question is discussed further in Section 8., Recommendations. Each class should also be highly homogeneous, that is the field from which the class was derived should map into the class. The field statistics furnished by MAXLIK enable the homogeneity of the classes to be studied in detail.

Several iterations were required to arrive at a set of classes suitable for classifying the Meyer subregion. In general classes corresponding to bare rock (90) and to the shrub mixture 50 were quite homogeneous, their training fields classifying well over 90% into the respective classes. On the other hand, classes corresponding to the grass-shrub mixtures 16 and 30 exhibited considerable confusion, and most of the effort expended at this stage was directed toward finding a suitable combination of classes into a new class which could capture these varieties upon maximum likelihood classification. No completely mechanical description can be given for this process, in which the analyst's knowledge of the study area plays as large a role as the derived statistics. After several tries a set of classes was arrived at for a final classification of the Meyer area.

2.3.2 Maximum Likelihood Classification

A classification of the Meyer area of the August 31 scene for the presentation of results was performed via MAXLIK, using the set of classes obtained by the process described in the preceding Section. Several iterations were required to select a set of pseudo-colors to associate with the respective classes. No precise description of the process of color assignment can be given. In general, colors assigned to classes which are to be distinguished should contrast sharply. The colors assigned to classes making up small percentages of the image should be bright and of a light shade (for example yellow) in order that they be noticeable and not lost in the background of the prevalent classes.

Once a suitable color selection was made for the Meyer subregion and a satisfactory class image obtained, attention was directed toward the Dona Ana subregion. Here it was a matter of selecting class types represented in the region, and then determining whether for a given type, say 20, to use a class defined by a field in the Meyer subregion, or to define a new class by a field lying in the Dona Ana subregion. Ultimately, four classes were carried over intact, two classes were defined by new fields, and two classes not included in the Meyer classification were used in the Dona Ana classification. Color selection was based on the Meyer selection.

The classification of the same subregions of the March 22 image proceded along the same lines. The August 31 field-file was carried over to the March 22 image, and it was determined by the "previously defined fields" overlay capability of FIELDEF that, with only two exceptions, the August 31 fields could be used to define March 22 classes. In those two cases, new fields referenced to the March 22 image were defined along with one additional field for a class not found in the August 31 image. Classes were generated from the March 22 composite image (S2AB), and the classification proceeded like the August 31 image.

2.4 RESULTS OF THE EXPERIMENT

A detailed analysis of the classification of the Fort Bliss scene will be presented in an ETL report being prepared by M. B. Satterwhite. The findings of the analysis can be summarized as follows:

a. Land cover classification from analysis of the Landsat data was primarily to a Level III and in some instances Level IV, using the Anderson land use/land cover classification system (Ref. 10). These classification levels and meaningful results could not have been obtained without an adequate ground truth data base.

b. Supervised classification discriminated two land cover classes on the alluvial fans. In the Meyer subscene the land cover classes corresponded well to the upper alluvial fans units, on which Larrea tridentata communities were almost exclusively found, and the mid and lower alluvial fans units on which Larrea-Grass, Flourensia-Grass and Hilaria-Scleropgon communities were found. Discrimination between the latter communities was not practical on this imagery because of pixel size, imagery scale, and the small area of some communities.

c. Differentiation between the grass communities described for the study area was achieved by comparing the March and August Landsat images for the same areas interpreting the land form conditions. Seasonal differences, observed primarily in band 7, permitted a greater differentiation between the cool and warm season grass communities and the various shrub-grass communities containing these grass species.

d. Confusion was found between some land cover mapping units on the same Landsat scene. Confusion was most important between some grass communities that had dark tones and occurred in the depressions of the basin area, and the shadows and other dark tone areas of the mountainous areas.

e. The land cover units of the limestone mountain areas and the adjacent upper alluvial fans were often confused with each other, which probably resulted from the sparse vegetation cover in these areas. Similarity of the ground surface materials, bedrock and gravels weathered from these rocks, probably accounted for much of the similarity between these land cover classes and the confusion among them during statistical classification.

f. Land cover units in the Hueco and Tularosa basins were discrete statistical units compared to land cover mapping units on the alluvial fans and rock outcrop areas. The major basin land cover unit, number 50, which occurred through most of the basin area, was not confused with any other land cover unit. The other land cover units of the basin, numbers 10, 14, 15, 52, 60, and 61 were confused among themselves which could be expected since the species comprising these communities varied in their community roles as either dominants, subdominants or associate species. The discreteness of land cover unit 50 from other basin communities probably lies with the large percentage of bare ground in the field of new primarily sandy soils. The other basin plant communities have more vegetation cover and less bare soil, also sandy textured soil. The confusion among the other communities probably is related to the varying percentage of grass cover in these grass, grass-shrub and shrub-grass communities. The greater grass cover darkens the spectral tone for the community, which leads to some uniformity and confusion among these communities.

Ratioed images did not prove to be useful in land cover classification. A variety of composite ratioed images with various color assignments were tried, but in all of them vegetative areas were more difficult to discern and identify than in the standard false color composite image, that is, Band 4 - Red, Band 5 - Green, Band 7 - Blue. Evidently the transformation of the albedo which makes ratioing so suitable for the identification of geological configurations makes vegetative configurations less distinquishable.

Overall, the experiment demonstrates the utility of the classification PMs developed for the DIAL system. Land cover classification to Anderson's Level III can be achieved provided adequate ground truth is available. The experiment pointed out some minor modifications to the FIELDEF and MAXLIK PMs which could be made to make their PMs more convenient for the analyst. These suggested modifications are discussed in Section 8, Recommendations.

Section 3

KARHUNEN - LOÈVE TRANSFORMATION

3.1 MOTIVATION OF THE KARHUNEN - LOÈVE TRANSFORMATION

Analysis of multichannel (multispectral) images such as those produced by Landsat MSS sensors has shown rather high correlation between the channels (bands) and therefore that a significant amount of redundancy exists among them. Further, the large quantities of data associated with multichannel (multispectral) images make it desirable if not imperative to increase the efficiency of the computations in the classification process. Finally, it is usually advantageous in feature extraction to use bands with as little correlation among them as possible. These facts taken together motivate the development of the Karhunen - Loève (K-L) transformation.

Consider an N-channel image. Each of its pixels X can be considered to be a point in an N-dimensional vector space. The fact that several of the channels are correlated means that it should be possible to find a transformation Y=T(X) of the image such that the first M components of Y, where M is significantly less than N, have substantially the same amount of information as X. If T is restricted to be an orthogonal transformation (which implies that it is linear) in order to preserve the properties of the original image, then all such T should be compared with respect to some criterion in order to choose the <u>optimum</u> orthogonal transformation.

A natural criterion is that the mean squared error between the original pixel X and the pixel obtained by reconstruction from the first M components of Y, summed over all the pixels in the image, be a minimum. The optimum orthogonal transformation under this error criterion is the Karhumen - Loève (KL) transformation. It is not hard to show (see Ref. 4 pp. 200-209), that

i) The rows of T are the eigenvectors of the covariance matrix C of the original image.

ii) The covariance matrix K of the transformed image is diagonal, therefore the channels of the transformed image are uncorrelated.

iii) The mean-squared error is the sum of the eigenvalues corresponding to the unused components of Y. But the eigenvalues are the variances of the components of Y, so if Y is ordered so that the first M components correspond to the M largest variances, the transformed image will have the minimum mean-squared error among all M-dimensional images resulting from an orthogonal transformation of the original image.

In the case of the four channel LANDSAT-2 MSS images, the transformed image typically has 95% of the variance in its first two channels, so that the computation necessary to produce a maximum likelihood classification of a LANDSAT-2 secne can be cut in half by the use of the K-L transformation, once the transformed image has been obtained.

3.2 COMPUTATION OF THE TRANSFORMED IMAGE

To obtain the transformed image, the original image is assumed to be a collection of N-dimensional vectors $\underline{x}_{ij}$, where N is the number of channels (bands) in the image, i=1,2,..., L is the line index, and j=1,2,..., S is the sample index of the pixels in the image. The variance-covariance matrix C of the image is an N x N symmetric positive definite (strictly speaking, it can only be assumed that C is positive semi-definite, where some components of the pixel vectors may be lineally dependent on the others. For original images, however, it is safe to assume that C is positive definite) matrix whose element C_{rs} is the r-th row and s-th column is given by

$$C_{rs} = \frac{1}{LS - 1} \left[\sum_{i=1}^{L} \sum_{j=1}^{N} (x_{ij} - \bar{x}^{r})(x_{ij} - \bar{x}^{s}) \right] \qquad (3.2\text{-}1)$$

where

$$\bar{x}^{r} = \frac{1}{LS} \sum_{i=1}^{N} \sum_{j=1}^{N} x^{r}_{ij} \qquad (3.2\text{-}2)$$

and where x^{r}_{ij}, r=1,2,..., N is the r-th component of the pixel vector $\underline{x}_{ij}$. Note that each element of C requires LS operations (an operation is one add and one multiply) and since C is symmetric N(N+1)/2 elements must be computed. The computation of C may therefore require an appreciable amount of time depending on the size of the image and the computer performance. If the K-L transform is being used to reduce the computation time for multispectral classification it is essential that an estimate of the time for the K-L transformation be made and compared with an estimate of the time saved in classifying the transformed image. In this connection it should be kept in mind that once the K-L transformed image has been obtained, it can be used for all subsequent classifications of that image, with the savings of computation time realized in each classification.

Since C is symmetric positive definite, its eigenvalues λ_k, k=1,2,...,N will be positive and its eigenvectors g_k will be orthogonal, and they can always be assumed to be normalized. The matrix G, whose element

$G_{k\ell} = (g_k)_\ell$, $k,\ell = 1,2,..,N$, is the matrix which brings C into diagonal form by a similiarity transformation. That is

$$GCG^{-1} = GCG^T = K \tag{3.2-3}$$

where $K = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_N)$. The K-L transformed image therefore is obtained from the original image by the transformation

$$y_{ij}^k = \sum_{\ell=1}^{N} G_{k\ell}\, x_{ij}^\ell \tag{3.2-4}$$

where y_{ij}^k , $k=1,2,\ldots, N$, is the k-th component of the pixel in the i,j-th position of the transformed image. It is a straight forward exercise to verify that K, the matrix of the transformed image, is diagonal. In fact, assuming for simplicity that $\bar{x}^r = 0$,

$$K_{rs} = \frac{1}{LS-1} \sum_{i=1}^{L} \sum_{j=1}^{S} y_{ij}^r\, y_{ij}^s \tag{3.2-5}$$

Substituting equation (3.2-4)

$$K_{rs} = \frac{1}{LS-1} \sum_{i=1}^{L} \sum_{j=1}^{S} \sum_{k=1}^{N} G_{rk}\, x_{ij}^k \sum_{\ell=1}^{N} G_{s\ell}\, x_{ij}^\ell \tag{3.2-6}$$

$$= \sum_{k=1}^{N} \sum_{\ell=1}^{N} G_{rk}\, G_{s\ell} \left(\frac{1}{LS-1} \sum_{i=1}^{L} \sum_{j=1}^{S} x_{ij}^k\, x_{ij}^\ell \right)$$

by a rearrangement. But the expression in parentheses is, by equation (3.2-1), $C_{k\ell}$. Further, since G is an orthogonal matrix, $G_{s\ell} = G^{-1}_{\ell s}$, where $G^{-1}_{\ell s}$ is the ℓ, s-th element of the matrix inverse to G. Therefore,

$$K_{rs} = \sum_{k=1}^{N} \sum_{\ell=1}^{N} G_{rk}\, C_{k\ell}\, G^{-1}_{\ell s} \tag{3.2-7}$$

which is equation (3.2-3) in component form.

If the bands of the transformed image are renumbered in order of decreasing eigenvalue magnitude, then the bands will also be ordered in decreasing information content. This is because the eigenvalues are the respective variances of the image bands, and the greater the variance, the greater the content of information in that band.

3.3 RADIOMETRIC ADJUSTMENT

Since the transformed image, like the original image, is usually intended to be displayed, it is necessary to radiometrically adjust the intensity values. For most purposes, linear transformations are suitable, in which case the pixel vectors in the transformed image would be given by

$$z_{ij}^{k} = a_k \, y_{ij}^{k} + b_k \tag{3.3-1}$$

where z_{ij}^{k} , k=1,2,...,N is the intensity of the k-th band of the i,j-th pixel.

In practice this transformation is combined with the K-L transformation, so the combined transformation takes the form

$$Z_{ij}^{k} = a_k \sum_{\ell = 1}^{N} G_{k\ell} \, x_{ij}^{\ell} + b_k \tag{3.3-2}$$

The parameters a_k and b_k are determined according to the statistical properties (radiometric enhancement) the transformed data is to have. In the KLTRAN program described in Section 7.3, four radiometric enhancement options are available to the user.

In all cases, the means μ_k of the intensity values in each of the bands of the transformed image are placed at the midpoint of the range of intensities available in the display. Since

$$
\begin{aligned}
\mu_k &= \frac{1}{LS} \sum_{i=1}^{L} \sum_{j=1}^{S} z_{ij}^{k} \\
&= \frac{1}{LS} \sum_{i=1}^{L} \sum_{j=1}^{S} \left(a_k \sum_{\ell=1}^{N} G_{k\ell}\, x_{ij}^{\ell} + b_k \right)
\end{aligned}
\tag{3.3-3}
$$

by equation (3.3-2), a rearrangement gives

$$
\begin{aligned}
\mu_k &= a_k \sum_{\ell=1}^{N} G_{k\ell} \left(\frac{1}{LS} \sum_{i=1}^{L} \sum_{j=1}^{S} x_{ij}^{\ell} \right) + b_k \\
&= a_k \sum_{\ell=1}^{N} G_{k\ell}\, \bar{x}^{\ell} + b_k
\end{aligned}
\tag{3.3-4}
$$

where $\bar{x}^{\ell}$, $\ell = 1,2,\ldots,N$ is the mean value of the intensities in the ℓ-th band of the original image. The requirement that μ_k be some prescribed value then means that a_k and b_k are connected by the relation

$$
b_k = \mu_k - a_k \sum_{\ell=1}^{N} G_{k\ell}\, \bar{x}^{\ell}
\tag{3.3-5}
$$

For an 8-bit display, for example, $\mu_k = 127.5$. The values of the a_k then depend on the desired contrast enhancement. Four options have been made available to the user of the KLTRAN program of Section 7.2.

Option 1.

No contrast enhancement.

This imples that $a_k = 1$, $k=1,2,\ldots, N$

Option 2.

Scaling to reduce the increase in interval length due to the K-L transformation.

This is accomplished by setting $a_k = 1/\sqrt{N}$, k=1,2,...,N, on the grounds that the transformation equation (3.2-4) combines N bands and therefore measures intensity values by a multiplication factor of the order of $\sqrt{N}$.

Option 3.

The distribution of intensities in each band is scaled so that H standard deviations are included in a interval of length d on each side of the mean, where H is user selected and d is the standard deviation of the unadjusted data. In this option the ratios among the variances of the principal components are no longer maintained.

This adjustment is accomplished by setting

$$a_i = \frac{d}{H\sqrt{\lambda_i}} , \qquad i = 1, 2, . . , N$$

Option 4.

The first principal component is radiometrically enhanced as in option 3, while the other principal components are enhanced proportionally.

Here $$a_i = \frac{d}{H\sqrt{\lambda_1}} , \qquad i = 1, 2, . . ., N$$

Section 4

RELAXATION METHODS

In the most direct application of the classification of multi-channel images by supervised algorithms of the maximum likelihood type or unsupervised algorithms of the ISODATA type, all of the channels contain observations of the same general nature. For example, the four channels of a LANDSAT image contain intensities of four bands of the electromagnetic spectrum from visible light to the near infra-red. However, there is nothing in the algorithms which impose a constraint on the image data, and it is perfectly possible, for example, to add a fifth channel to a LANDSAT scene containing the average elevation data, and classifying the resulting five-channel image with a maximum likelihood algorithm. Then if there were two different crop types, say, which had somewhat similiar spectral signatures but each of which was found only in a narrow range of altitudes, the accuracy of the five-channel classification which takes into account elevation data would be expected to be greater than the accuracy of the four-channel classification.

Directly adding channels to a multi-channel image is one way to take into account additional information in order to improve classification accuracy. Another way is to impose constraints which are formulated from additional information to prevent or to make unlikely an incompatible classification. A class of algorithms to do this, known as "relaxation" methods, has been under development in recent years (see Ref. 2 and its references for a fuller discussion). Relaxation methods (the term may have been suggested by the class of iterative methods for solving a system of linear equations by reducing the largest residual in each iteration) interatively go through all of the pixels to refine the class assignments (label assignments) by taking into account compatibility restraints to remove ambiguous or improbable labeling. As part of the study two relaxation algorithms were developed. One was selected for coding and testing. The other was to be coded by ETL, but this effort was not completed at the time of preparation of this report.

4.1 ALGORITHM DESCRIPTION

The two algorithms developed for modifying class assignment assume that an initial classification has been performed, and that each pixel has been assigned N labels, each with a probability $p_i \geq 0$ with $\sum_i p_i = 1$.

The algorithm described in Section 4.1.1 looks for boundaries between aggregates of similiar pixels. If the pixel is an interior one, the probability that it is in the same class as its neighbor is increased. If it is a border pixel, either the probability that it is one of the two neighboring classes is increased, or it is assumed to belong to neither of the neighboring classes (for example it might be on a road between two agricultural fields) depending on certain tests. The algorithm described in Section 4.1.2 is based on concepts developed by A. Rosenfeld and his colleagues at the University of Maryland Computer Science Center.

4.1.1 Boundary Selection Relaxation Algorithm

This relaxation process for scene labeling begins with a set of probabilities $\{p_{ij}(\lambda_1),\ldots,p_{ij}(\lambda_m)\}$ for pixel (i,j). This set of probabilities is derived from the maximum likelihood classification function. The $p_{ij}(\lambda_k)$ are normalized such that

$$\sum_{k=1}^{M} p_{ij}(\lambda_k) = 1$$

where M is the number of labels/categories stored for each pixel. The range of k is N, which is $\geq$M; that is N is maximum number of categories that could be associated with a pixel.

The general notation used for identifying pixel neighbors is given in Figure 4.1-1. The offset shown is 3; that is, 48 neighboring pixels contribute to the labeling of a pixel. The actual offset to be used in the process is yet to be resolved, but for this description three will be used.

The basic assumption for this process is that scene areas consisting of many pixels are relatively homogeneous and labeled similiarly. One of the concerns in updating pixel labels under this basic assumption is whether or not a pixel is influenced by a boundary. The process begins with the determination of the existence of a boundary.

4.1.1.1 Boundary Existence Determination

Four boundary situations will be considered; upper and lower, right and left, and the two diagonal cases. (See Figure 4.1-2).

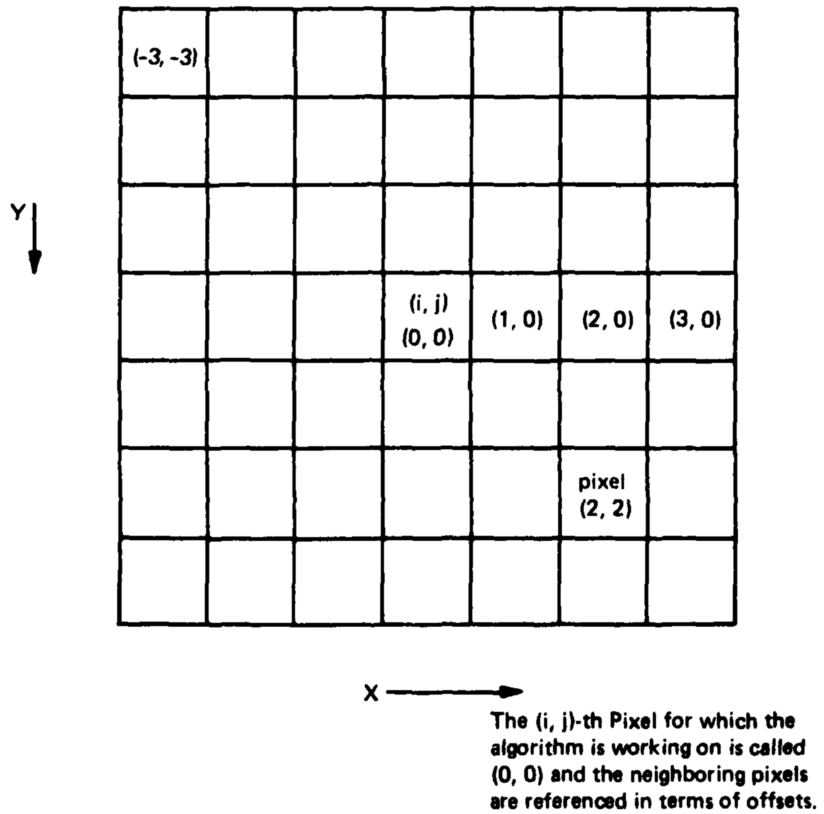


Figure 4.1-1. Relative Pixel Coordinate Notation

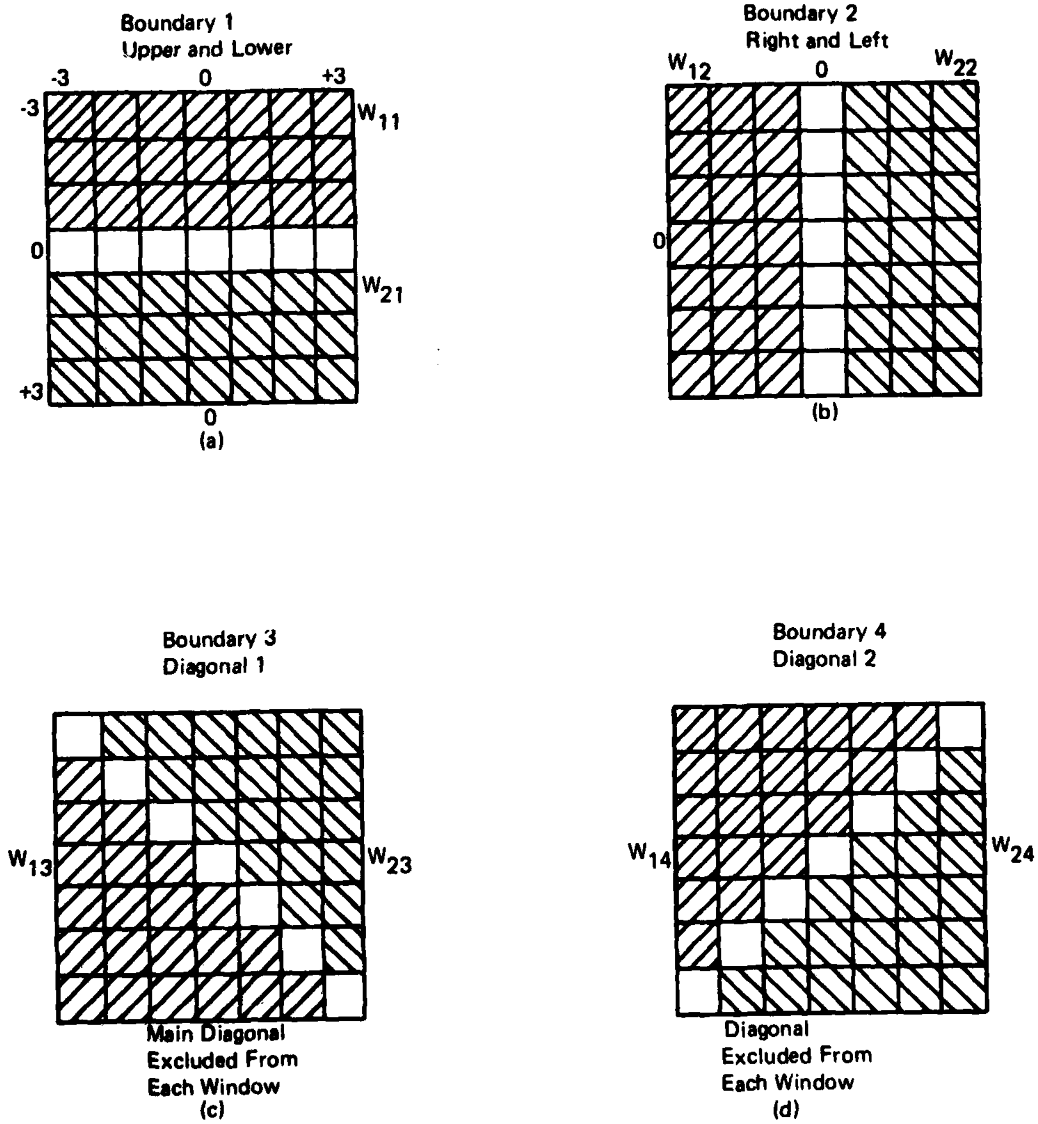


Figure 4.1-2. The Four Boundary Possibilities

Associated with each boundary is a pair of pixel windows, whose size is a function of the selected offset. For example, under the given conditions the window W_{11} associated with the upper part of boundary 1 includes columns -3 to 3 and rows -3 to -1 for a total of 21 pixels. (See Figure 4.1-2a). Existence of a boundary will be based upon the H_0 hypothesis (equal means) for corresponding window signatures. The signatures, means and variances for the above window are given below

$$P_{11}(\lambda_k) = \frac{1}{21} \sum_{j=-3}^{-1} \sum_{i=-3}^{3} p_{ij}(\lambda_k), \ k = 1, N$$

$$\sigma_{11}^{2}(\lambda_k) = \frac{1}{20} \sum_{j=-3}^{-1} \sum_{i=-3}^{3} (p_{ij}(\lambda_k) - P_{11}(\lambda_k))^2, \ k = 1, N$$

Corresponding window signatures are evaluated using a Hotelling T^2 measure to test the H_0 hypothesis (equal means and variances).

If the hypothesis H_0 from the four boundaries is true (no boundary) then the update will be based upon the pixels within a window centered about the (0,0) pixel. First, the case when a boundary influences the pixel of interest is considered.

4.1.1.2 Boundary Present

This subsection is concerned with the case in which it has been determined that one of the four boundary conditions (figure 4.1-2) has been detected. In this situation if the pixels in the window centered at (0,0) were to be used in updating the probability vector for the center pixel, the effect of the boundary would be smoothed or obliterated. Accordingly, the first question to be answered is whether or not the center pixel is a member of either side of the boundary, or belongs to the boundary itself. A sum of squares criterion * is suggested for resolving this question. One computes

$$S_{ij} = \sum_{k=1}^{N} (p_{00}(\lambda_k) - P_{ij}(\lambda_k))^2, \ i = 1, 2$$

where j is the boundary indicator. The minimum S_{ij} is compared to a threshold T. If the minimum is less than T the pixel is said to belong on the side of the boundary corresponding to the minimum S_{ij} and will be updated as a member of this area. On the other hand if the minimum exceeds the threshold then the pixel will be considered to be a member of the boundary itself.

Next based upon the above results one of two options will be selected.

Option 1 Member of Boundary

This is an interesting case in that the algorithm states that a boundary/ discontinuity exists but the labels of pixel (0,0) are not like any of the neighboring windows. It seems that this does not really add any information about what the labels should be and it would seem reasonable that the current set of labels and accompanying probabilities should remain the same as the estimates of the prior iteration. On the other hand an argument could be formulated for incrementing the probability estimate associated with the most probable label. We propose that they stay the same.

*It may be advisable to use a normalized sum of squares criterion. Also for this test the $P_{ij}(\lambda_k)$ are normalized.

Option 2 Member of a Window

Once the particular window to which this pixel belongs has been determined, the following steps will be performed to update the estimates.

1. Calculate a mean vector for a subset of pixels in the selected window, say six pixels. As an example if window 1 1 was selected then pixels

(-1, -2)	(0, -2)	(1, -2)
(-1, -1)	(0, -1)	(1, -1)

would be used in the computation:

$$P(\lambda_k) = \frac{1}{6} \sum_{j=-2}^{-1} \sum_{i=-1}^{1} p_{ij}(\lambda_k), \; k = 1, N$$

2. Order the $P(\lambda_k)$ by descending probability and normalize so the first M add to one.

3. For the λ_k that gave the maximum probability increase the corresponding probability of pixel (0,0) by

$$p'_{00}(\lambda_k) = p_{00}(\lambda_k) + P(\lambda_k) \left(1 - \left| P(\lambda_k) - p_{00}(\lambda_k) \right|\right)$$

4. Normalize the $p'_{00}(\lambda)$

5. Order in descending probability.

It is possible, but not likely, that the λ_k corresponding to the maximum probability of P (λ) may not be in the list of P_{oo} (λ). Under this situation the label corresponding to the least probable is replaced by λ_k and the computation continues at 3. Also it should be noted that the updating equation given in 3. needs to be evaluated and tuned experimentally.

4.1.1.3 No Boundary Present

When no boundary is present the assumption is that the center pixel is like its neighbors, so a mean vector is computed from its neighbors. As a first suggestion a 3 x 3 window excluding the center pixel would be used:

$$P(\lambda_k) = \frac{1}{8} \sum_{j=-1}^{+1} \sum_{\substack{i=-1 \\ i \neq j \neq 0}}^{+1} P_{i,j}(\lambda_k), \quad k = 1, N$$

The processing then proceeds as given in Option 2 of Section 4.1.1.2 (Member of a Window), step 2.

4.1.2 The Relaxation Method of Rosenfeld et Al

The relaxation process described in this section is based on the approach developed by A. Rosenfeld and his associates at the Computer Science Center of the University of Maryland.

As in the previous section, the pixels in the scene are labeled (i,j) where i = 1,2..., (number of lines) and j = 1,2..., (number of samples). Associated with each pixel is a set of labels λ_k, k = 1, 2..., N, and associated with each label is a probability $p_{ij}(\lambda_k)$ such that $0 \le p_{ij}(\lambda_k) \le 1$

and

$$\sum_{k=1}^{N} p_{ij}(\lambda_k) = 1$$

for all pairs (i,j).

The entropy H_{ij} for the (i,j)th pixel, which is a measure of the information associated with the pixel, is defined as

$$H_{ij} = - \sum_{k=1}^{N} p_{ij}(\lambda_k) \ln p_{ij}(\lambda_k) \qquad (4.1\text{-}1)$$

Let p_{ij} be the N vector whose components are the $p_{ij}(\lambda_k)$. An entropy should have the property that it increases as the information content decreases. In the case of entropy as defined by (4.1-1) it attains its maximum when the λ_k are equally likely, that is $p_{ij}(\lambda_k) = 1/N$, and attains its minimum = 0 when $p_{ij} = (0,\ldots 0,1,0,\ldots,0)$, that is the (i,j) -th pixel has label λ_m with probability one. The proposed relaxation scheme has the property that it decreases the entropy of each pixel, and in fact the scheme can be shown to converge to the condition in which each pixel has a definite label, say λ_m, with probability 1 and all other labels with probability 0.

4.1.2.1 Algorithm Formulation

The basic formula for updating the probability vector p_{ij} associated with the (i,j)-th pixel is

$$p_{ij}^{(n+1)} \lambda_k = \frac{p_{ij}^{(n)}(\lambda_k)\left[1+q_{ij}^{(n)}(\lambda_k)\right]}{\sum_{k=1}^{N} p_{ij}^{(n)}(\lambda_k)\left[1+q_{ij}^{(n)}(\lambda_k)\right]} \qquad (4.1\text{-}2)$$

where

$$q_{ij}^{(n)}(\lambda_k) = \sum_{l,m} C_{ij;\ell m} \sum_{k'} r_{ij;\ell n}(\lambda_k, \lambda_{k'})\, p_{\ell n}^{(n)}(\lambda_{k'}) \qquad (4.1\text{-}3)$$

In (4.1-2) and (4.1-3) the superscript n indicates the iteration number; $r_{ij;\ell m}(\lambda_k, \lambda_{k'})$, which take on values in the interval $[-1,1]$, is the measure of the compatibility of label λ_k for pixel (i,j) with the label λ_k' for pixel ℓ, m; and $C_{ij;\ell m}$ is the weighting of the pixels (ℓ,m) which are neighbors of pixel (i,j), and it is required that $0 \leq C_{ij;\ell m} \leq 1$ and $\sum_{\ell m} C_{ij;\ell m} = 1$ for each pixel (i,j).

In theory r's and C's could be defined for each pixel, the C for a given pixel could be defined for all neighbors, and the r's could take on any value between -1 and +1. For purposes of investigating the utility of the relaxation method in refining the labeling of a scene classified by the maximum likelihood supervised classification method (class maps) it is better to simplify the definition of the C's and the r's as follows:

$C_{ij;kl}$ will be independent of ij. For any interior pixel, which can be considered to have the coordinates (i,j) = (0,0), $C_{k\ell}$ is inversely proportional to the Euclidean distance between (0,0) and (k, ℓ). This scheme excludes the center pixel from the averaging process.

To simplify the compatability matrix, $r_{ij;k\ell}(\lambda_k, \lambda_{k'})$ will be considered independent of both (i,j) and (k, ℓ), and will take on only the values +1 if labels λ_k and $\lambda_{k'}$ are highly compatible, 0 if they are independent, and -1 if they are highly incompatible. In addition $r(\lambda_k, \lambda_{k'}) = r(\lambda_{k'}, \lambda_k)$ and $r(\lambda_k, \lambda_k) = 1$, that is r is symmetric.

In practice the user will supply the values of r which in the case of class maps will depend on how likely two classes are to be found in adjoining pixels.

With these definitions, (4.1-3) takes the form

$$q_{ij}^{(n)}(\lambda_k) = \sum_{\ell,m} C_{ij;\ell m} \sum_{k'=1}^{N} r_{ij;\ell m}(\lambda_k, \lambda_{k'}) p_{\ell m}^{(n)}(\lambda_{k'}) \qquad (4.1\text{-}4)$$

This algorithm can be shown to converge linearly to a unique *solution* for a number of cases of practical interest.

4.1.2.2 Accelerated Convergence

Linear convergence is usually not rapid enough for practical applications. A modified form of (4.1-2) which converges in only a slightly smaller number of cases but which converges geometrically is given by

$$p_{ij}^{(n+1)}(\lambda_k) = \frac{\left[p_{ij}^{(n)}(\lambda_k)\right]^{\alpha_1}\left(1 + q_{ij}^{(n)}\right)^{\alpha_2}}{\sum_{k=1}^{N}\left[p_{ij}(\lambda_k)\right]^{\alpha_1}\left(1+q_{ij}^{(n)}\right)^{\alpha_2}}$$

where α_1 and α_2 are integers. In practice one would set $\alpha_1 = \alpha_2 = \alpha$.

The optimum value of α depends on the particular problem but in general it can be said that as α increases the required number of iterations decreases; however, the danger of erratic behavior of the algorithm because of roundoff increases. A value of $\alpha = 2$ or 3 is a reasonable choice for the first attempt at labeling a new scene.

4.2 TESTING THE ROSENFELD ALGORITHM

As mentioned in Section 4.1, it was agreed that ETL would code the boundary detection algorithm described in Section 4.1.1, and that IBM would code the algorithm described in Section 4.1.2. Since the coding of the boundary detection algorithm was not yet completed when this report was prepared, only the implementation of the Rosenfeld algorithm was tested.

Application of the Rosenfeld algorithm requires the sequential use of three DIAL PMs developed for that purpose, PLABEL, Section 7.5; RELAX, Section 7.6; and ITRES, Section 7.7. PLABEL assigns classes and corresponding probabilities to each pixel of a multi-channel image by the same maximum likelihood (Bayes) classifier as MAXLIK. The significant difference is that the MAXLIK assigns the most probable class (label) to each pixel and produces a two channel output image which has the same number of lines and samples as the input image and in which the first channel is the class ID and the second channel is the chi-squared value. PLABEL produces a multi-channel (NLABEL dimensions), 15 bit output image which has the same number of lines and samples as the input image, and in which the first five bits of a given channel are the class ID and the second ten bits the corresponding probability (multiplied by 1023). RELAX has as its input the multi-channel class image to which it applies the Rosenfeld algorithm, producing an output image of the same type. ITRES allows class maps to be displayed on the COMTAL, and summary statistics to be displayed on the Tektronix.

In the test of the algorithm, the LACIE intensive site used in Ref. 1. was the image to be classified. A set of six classes derived earlier, CORN, SPWH (spring wheat), OATS, and three pasture grasses GRPS501, GRPS502, and GRPS504 was selected for PLABEL. The correctness of the classification was confirmed by comparison with a previous classification of the same scene. The class map, named PLABELLIS4, was the input to RELAX. The input parameters were selected as follows:

MXLABEL (number of classes)	= 6
NLABEL (number of classes retained)	= 3
IBORDT (side of surrounding pixels array)	= 5
R (class compatibility	= identity matrix
C (weighting matrix)	= default (Euclidean distance)
ALPHA (covergence factor)	= 1

The resulting class map was checked by hand computing a specific output vector, corresponding to line 41, sample 50. The input data is shown in Figure 4.2-1, where in each box (pixel location) the first line is the probability vector, the second line is the corresponding class ID vector, and the number in parenthesis is the weighting factor. The results of the check were:

	RELAX			Hand Computation		
Output Probs.	0.8859	0.1019	0.0122	0.8864	0.1213	0.0123
IDs	6	2	4	6	2	4

In view of the fact that the original data had only two decimal places, the agreement to one unit in the third place can be considered confirmation that the RELAX PM is computing the Rosenfeld algorithm correctly.

(.35) .98 .02 .0 5 4 1	(.45) .70 .30 0 4 5 2	(.5) 1.00 0 0 4 5 2	(.45) .92 .05 .03 4 2 6	(.35) .54 .30 .17 6 4 2
(.45) .83 .17 .0 4 5 2	(.71) .76 .13 .11 4 6 2	(1.0) .58 .29 .14 6 4 2	(.71) .84 .09 .07 4 2 6	(.45) .47 .20 .25 6 2 4
(.5) .86 .12 .02 4 2 6	(1.0) .71 .29 .0 2 6 4	0 .88 .11 0 6 2 4	(1.0) .98 .01 0 4 2 6	(.5) 1.00 0 0 4 2 6
(.45) .73 .27 0 6 2 4	(.71) .55 .45 .0 6 2 4	(1.0) .72 .16 .11 4 2 6	(.71) 1.00 0 0 4 2 5	(.45) 1.00 0 0 4 2 6
(.35) 1.00 0 0 4 2 6	(.45) 1.00 0 0 4 2 6	(.5) 1.00 0 0 4 2 5	(.45) 1.00 0 0 4 5 2	(.35) 1.00 0 0 4 5 2

Figure 4.2-1. Input Data for Pixel (41, 50)

RELAX was applied successively to the class map produced by PLABEL. That is, the output class map from the first application (iteration) of RELAX became the input class map for the second application (iteration), etc. The results are summarized in Table 4.2-2. It can be seen that the average of the maximum (per pixel) probability is increasing monotonically (although the rate of increase is decreasing), and that the average image entropy is decreasing at a rate exceeding 20% per iteration. The respective standard deviations are also decreasing. Note that the standard deviation for the entropy is of the same order of magnitude as the mean of the entropy. This is because the pixel entropy is approximately exponentially distributed, and for an exponential distribution the mean and standard deviation are equal.

Table 4.2-2. Summary Results for RELAX

	Original Class Map	First Iteration	Second Iteration	Third Iteration
Max prob. mean	.892	.918	.937	.949
Max prob. std dev	.145	.133	.120	.110
Ave entropy mean	.259	.196	.152	.121
Ave entropy std dev	.265	.252	.233	.215

The effect of the relaxation algorithm is demonstrated visually in Figure 4.2-2. The class map produced by PLABEL, Figure 4.2-2(a) is to be compared with the class map after three iterations of RELAX, Figure 4.2-2(b). It is very apparent that the "speckle" (individual or very small clusters of pixels of one class scattered among the pixels of a different class) has to a large extent been removed. This is particularly apparent in the wide swath running from the Northwest to the Southeast corners of the image, consisting largely of pasture grasses with a relatively small number of regularly shaped cultivated fields

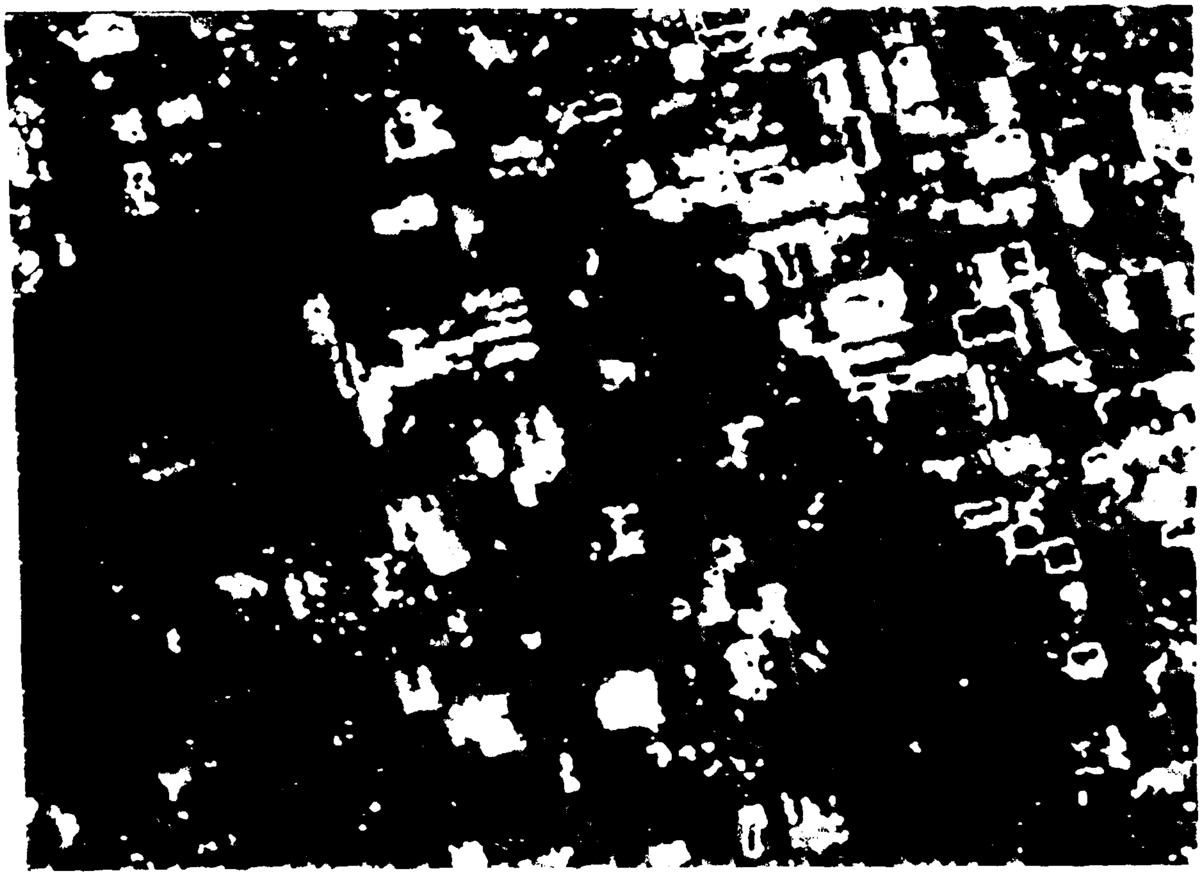


Figure 4.2-2(a) Relaxation Class Maps. Initial Class Map Produced by PLABEL

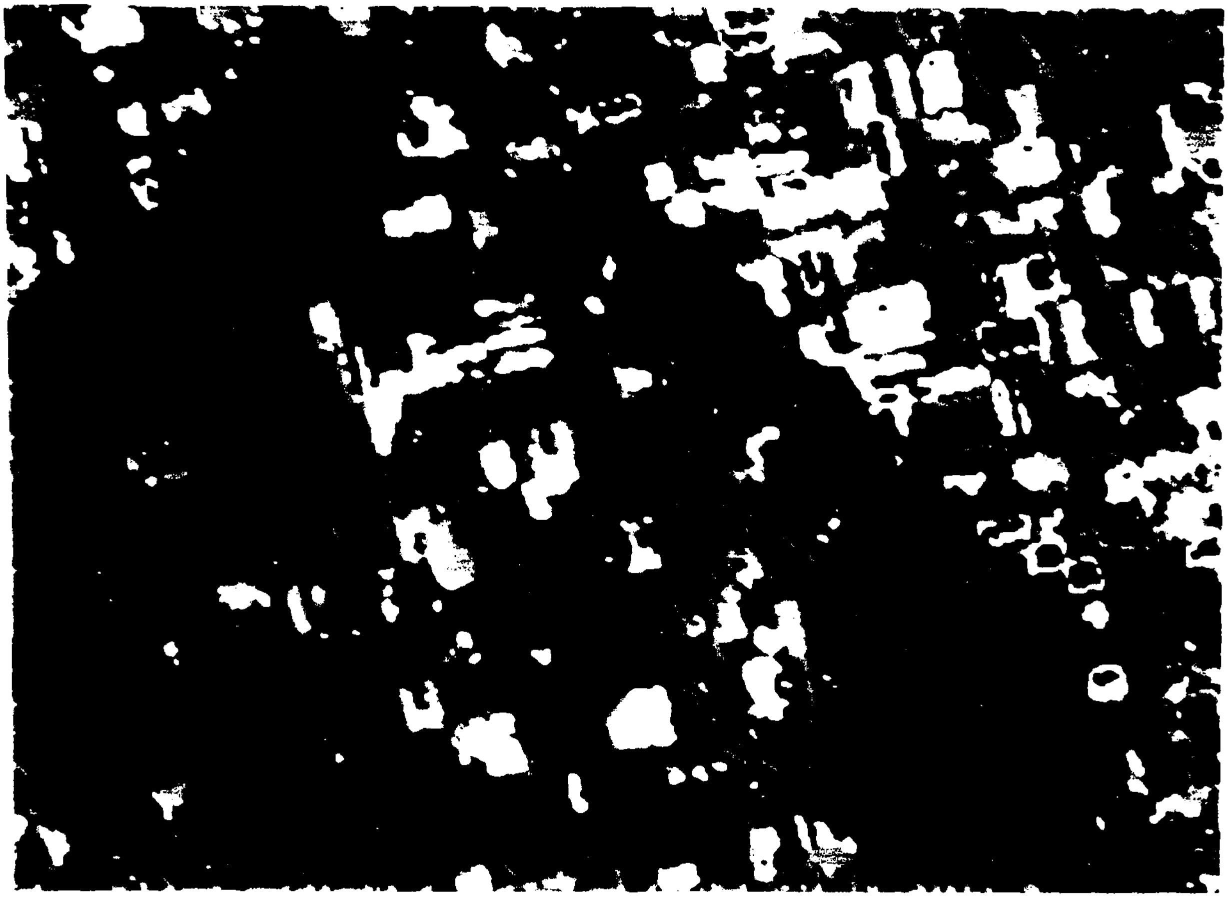


Figure 4.2-2(b) Relaxation Class Maps. Class Map After Three Iterations of RELAX

distributed over the region. Cultivated field boundaries have been sharpened as well. Since the final class map bears a greater similiarity to the ground truth than the initial class map, it can be said that relaxation has improved the accuracy of the maximum likelihood classification.

In order to determine the effect of the convergence factor ALPHA, an interation was performed with ALPHA = 2. The results were

Input average entropy = 0.116

Output average entropy = 0.048

Thus the entropy has been reduced by 50% per iteration as compared with 23% per iteration when ALPHA = 1. It can be concluded that when the iteration process is converging, as is the case here, doubling ALPHA doubles the rate of convergence, which is the behavior predicted by Reference 2.

4.3 CONCLUSIONS

Two relaxation algorithms for modifying the probability of class assignment based on the assignments of neighboring pixels and the compatibility of the various classes have been developed. One of the algorithms was coded as PM RELAX and tested on a classification of the LACIE intensive site.

The tests indicate that the algorithm performs as expected, increasing the average value of the maximum per pixel probability of the most likely class (ideally this value should be 1.0), and decreasing the average image entropy (ideally the value should be 0.0). Convergence is linear in entropy, somewhat slower in probability. The algorithm has a convergence factor which was set equal to 1 for most of the test runs. Increasing the factor to two doubled the rate of convergence, as predicted by the theory.

Further tests of the relaxation algorithm implemented in RELAX are clearly warranted, based on the encouraging results of the first set of tests. Other weighting matrices besides the default Euclidean weights should be tried, as well as other compatibility matrices besides the default identity matrix. In the tests a 5 x 5 array of neighboring pixels was used. A 3 x 3 array should be tested to determine if there is a trade off between computation time and algorithm effectiveness.

The second algorithm developed was not completed at the time the final report was prepared. Based on the positive results obtained with the first algorithm, it is strongly suggested that the implementation of the second algorithm be completed. In this connection it should be noted that the control structure and data flow can be carried over virtually intact. All that needs to be designed and coded is the subroutine which applies the algorithm to a given pixel, the input data being the probabilities (ordered in decreasing order of magnitude) and class IDs for that pixel, and the output being the revised probabilities and IDs. A more detailed discussion of how this implementation using RELAX can be done will be found in Section 7.6.

Section 5

MAXIMUM LIKELIHOOD CLASSIFICATION ON THE STARAN

5.1 MOTIVATION FOR THE TASK

Supervised classification is typically a computationally intensive function.

For example a maximum likelihood classifier (MLC) assigns each observation vector x to the class that minimizes

$$S = -2 \ln p_i + \ln |\Sigma_i| + (x-\mu_i)^T \Sigma_i^{-1} (x-\mu_i) \tag{5.1-1}$$

where p_i is the probability of observing the i^{th} class, μ_i is the mean of the i^{th} class and Σ_i is the covariance matrix of the i^{th} class. In order to evaluate S for a particular x there are $(n^2 + 3n)/2$ multiplications and $(n^2 + 3n)/2$ additions necessary for each class, where n is the dimension of observation vector. To relate this to computation time consider the following classification task; a 4 dimension observation vector, 10 classes, 2340 x 3240 vectors (7.58 million)*. This task takes approximately 200 minutes on an IBM 360/75 and 660 minutes on the CDC 6400.

The magnitude of these timing numbers, which makes the classification of large images impractical if not impossible led to the activity described in this section, which had the objective to reduce the MLC response time. Currently, MLC is performed on the CDC 6400 via the MAXLIK processing

* This corresponds to a full Landsat scene.

module. Our approach to improving the response time was to modify MAXLIK so that the STARAN could be used as an attached processor for the computation of S. By so doing it was anticipated that at least a 20 to 1 performance gain could be achieved for the above defined classication task. Also this type of computation is well suited for the STARAN because S for each vector can be performed in parallel and can be expressed as simple matrix operations.

Responsibility for this activity was divided between IBM and ETL. IBM did the formulation of the algorithm, accuracy analysis and developed the CDC 6400 software while ETL developed the STARAN software.

5.2 PROBLEM FORMULATION

In order to hold down field space and better utilize the STARAN architeture it was necessary to restructure the formulation of S from that given in equation 5.1-1. By taking into account that Σ_i is real, symmetric and positive definite it can be shown that Σ_i can be expressed as a product of an upper triangular matrix and its transpose:

$$\Sigma_i^{-1} = UU^T$$

With this formulation

$$\begin{aligned} \Sigma_i^{-1} &= (UU^T)^{-1} \\ &= (U^T)^{-1} U^{-1} \\ &= (U^{-1})^T U^{-1} \end{aligned}$$

which can be written as

$$\Sigma_i^{-1} = L_i^T \; L_i$$

where $L_i = (U^{-1})^T$ is a lower triangular matrix.

Now equation 5.1-1 can be written as

$$S = -2 \; \ell n \; p_i + \ell n \; |\Sigma_i| + (x-\mu_i)^T \; L_i^T \; L_i \; (x-\mu_i) \qquad (5.1\text{-}2)$$

$$= -2 \; \ell n \; p_i + \ell n \; |\Sigma_i| + (L_i x - L_i u_i)^T \; (L_i x - L_i \mu_i) \qquad (5.1\text{-}3)$$

which is just a constant plus the inner product of the vector

$Y=(L_i x - L_i \mu_i)$ with itself. (5.1-4)

From this formulation we can observe that the leftmost product in Y is both observation vector and class dependent and so is computed for each vector x, while the rightmost product is strictly class dependent and therefore need be computed only once per classification task. When using this formulation (equation 5.2-2 and 5.2-3) a different processing order is suggested from that suggested by equation 5.1-1. For equation 5.1-1 the order suggested is

(1) $\delta = (x-\mu_i)$

(2) $\chi^2 = \delta^T \; \Sigma_i^{-1} \; \delta$

$$= \sum_{j=1}^{n} \delta_j \sum_{k=1}^{n} \delta_k \; C_{jk}$$

where C_{jk} = elements of Σ_i^{-1}

(3) $S = \chi^2 - 2 \; \ell n \; p_i + \ell n \; |\Sigma_i|$

The latter formulation (that is Equation 5.2-2 and 5.2-3) suggests the following order

(1) $\delta = L_i x$

$$\delta_j = \sum_{k=1}^{n} \ell_{jk} x_k$$

(2) $Y = \delta - L_i \mu_i$

(3) $\chi^2 = \sum_{k=1}^{n} y_i^2$

(4) $S = \chi^2 - 2 \ln p_i + \ln |\Sigma_i|$

The importance of this latter formulation is that it holds down field space requirements as well as the field sizes used in the add and multiply operations. This will dramatically improve response time, because STARAN is a bit operation machine, which means that the larger the field size (more bits) the longer the execution time of the operation. As an example to multiply two 24 bit fields takes 3 times the execution time of multiplying a 24 bit field by a 8 bit field.

Table 5.2-1 summarizes the primary processing steps performed by the STARAN in order to support an MLC task. Other information dealing with MLC on the STARAN can be found in other sections of this report and a report to be published later. The list below gives the document and section in which each of these pertinent topics is discussed.

Topic	Location
1. Current 6400 MLC use and software	ETL-0172 Section 3.5
2. Control software on the 6400 for MLC on the STARAN and user information	This report Section 7.8
3. Detailed description of data flow and format used between the 6400 and the STARAN	This report Appendix A
4. STARAN software and field size description	Report to be published later by ETL

Step	Description	Input Data Description	Output Data Description
1	Develop B Vectors one per Class Done once per Classification Task $B_i = L_i \mu_i$	Element of L_i 24 bits sign bit 23 fractional Element of μ_i 16 bits sign 7 integer 8 fraction	Elements of B_i 32 bits sign 12 integer 19 fractional
2	Complement Most Significant bit of each element of x Vector	Element of x 8 bits 8 integer	Element of x 8 bits 1 sign 7 integer
	Repeat Steps 3 thru 8 for every class		
	Repeat Steps 3 thru 6 for each vector component		
3	Develop k^{th} Component of A Vector $A = L_i X$ $k = 1, n$	Element of x 8 bits 1 sign 7 integer element ℓ 1 sign 23 fractional	Element of a 31 bits 1 sign 11 integer 19 fractional

Table 5.2-1. STARAN Primary Processing Steps

Step	Description	Input Data Description	Output Data Description
4	Subtract b_k from a_k and normalize	As given above	17 bits sign 8 integer 8 fractional
5	Square difference $(b_k-a_k)^2$	As given above	34 bits sign 17 integer 16 fractional
6	Accumulate squared differences $\chi^2 = \sum_{k=1}^{N} (b_k - a_k)^2$ and normalize	As given above	38 bits sign 21 integer 16 fractional
7	Add constant $-2 \log \phi_i + \log \lvert\Sigma_i\rvert$ to χ^2	Constant 24 bits sign 7 integer 16 fractional	16 bits 8 integer 8 fractional
8	Compare with prior minimum and store minimum and ID	16 bits 8 integer 8 fractional	16 bits 8 integer 8 fractional 4 bits ID

Table 5.2-1. STARAN Primary Processing Steps

Section 6

COURSES AND DEMONSTRATIONS

One of the activities of the study was the conducting of demonstrations and courses in the feature extraction capabilities added to the DIAL system. The purpose of the demonstration, which were given to relatively large groups (25-30) of individuals from both within and outside the ETL community, was to acquaint a varied audience with concepts of multi-channel classification and their implementation on DIAL, and to stimulate suggestions for further development. The purpose of the course was to give a small group (5-6) of ETL members a more detailed introduction to the theory of multi-channel classification and "hands-on" experience with the DIAL supervised and unsupervised classification PMs.

6.1 DEMONSTRATION OVERVIEW

Demonstrations were given on Nov. 8, Nov. 27, and Dec. 2, 1979 in each case to about as large an audience as could comfortably view the four COMTAL displays. The demonstrations were formal in that they were conducted by the investigators with no direct audience participation, although questions were of course solicited.

The program for the demonstrations which lasted approximately 90 minutes was as follows:

(Before the audience entered the DIAL room a false color image had been put up on display A, and an expanded image of a portion of A with field boundaries put up on C. DIAL PMs used are in capital letters, COMTAL displays used are in parentheses after description).

- Introduction, nature of the IBM effort, brief introduction to the basic concepts of feature extraction and multi-spectral (multi-channel) classification

- Use of DSPSUBN to display images and subimages, GRYSI to adjust contrast (displays B and D)

- Indentification of area of interest in a large image (typically Landsat) with SCROLL

- Definition of area to be classified and of training fields and test fields (both areal and linear) with FIELDEF (B,D)

- Calculation of class statistics (mean vectors, covariance matrices) and Bhattacharyya distance between similiar and dissimiliar classes with CLASTAT

- Supervised classification of LACIE intensive site with MAXLIK. Selection of test fields, selection of classes, color assignments, display of results (Tektronix, B, D)

- Unsupervised classification of LACIE intensive site with CLUSTER. Starting parameters, starting vectors, review of iterations, display of results (Tektronix, B, D).

- Comparison of the supervised and unsupervised classification results with CHGDET (B).

- Dimensionality reduction of LACIE intensive site with KLTRAN. Band-by-band comparison of original and transformed image (D).

- Comparison of the supervised classification of original (four-band) and transformed (two-band) images with CHGDET (D).

- Comparison with DSP of false color display of bands 4, 5, and 7 of a Landsat image with ternary ratio image constructed with RATIOF (B,D)

6.2 CONTENT OF THE COURSE IN CLASSIFICATION

The one-day course in multi-spectral (multi-channel) classification on the DIAL system was given in two sessions. The morning session was a lecture on the principles of multi-spectral classification, while in the afternoon session students gained "hands-on" experience by working through a model classification problem.

6.2.1 Principles of Multi-spectral Classification

The lecture on the principles of multi-spectral classification was based on the material which has been presented at intensive short courses in remote sensing offered two or three times a year by the George Washington University Division of Continuing Engineering Eduction.

- Nature of multi-spectral (multi-channel) data
  Typical spectral signature (wheat) (Figure 6.2-1)

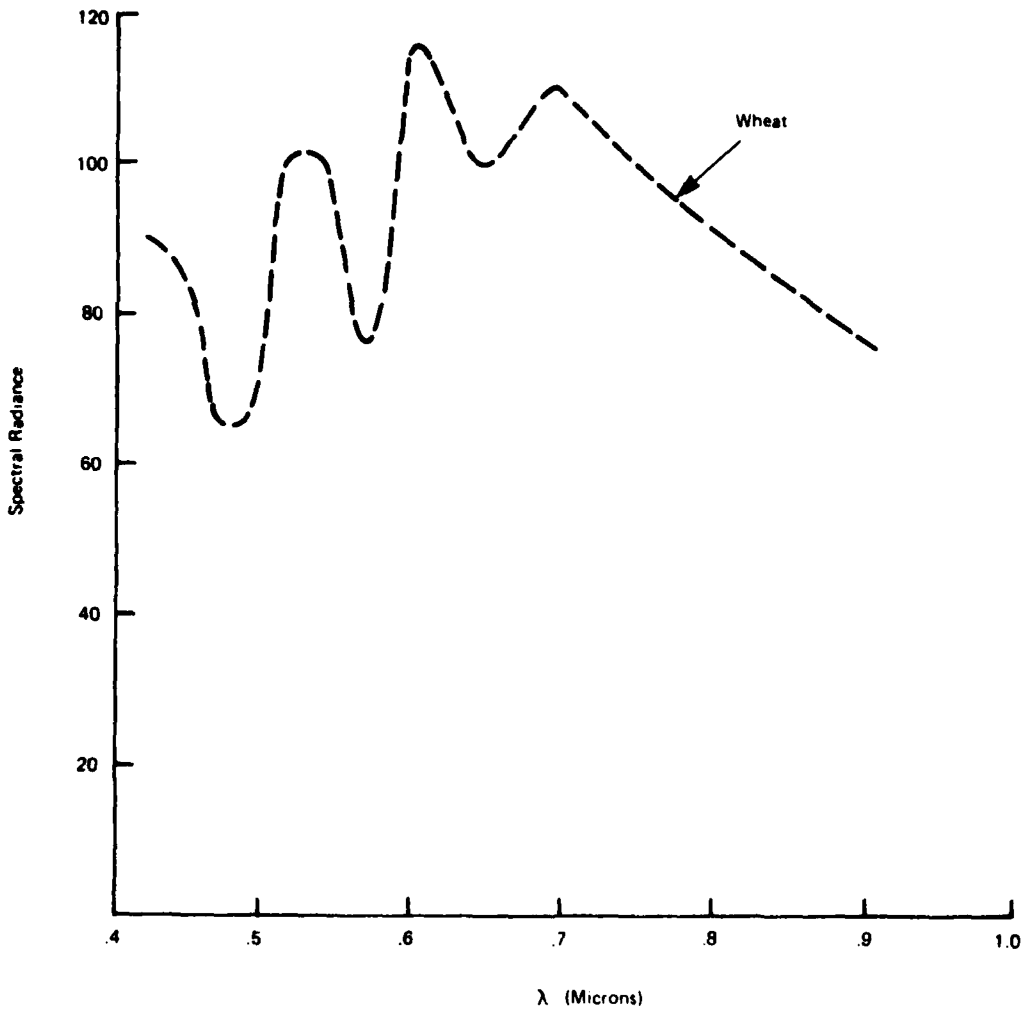


Figure 6.2-1. Typical Spectral Signatures. 7-Bit Spectral Radiance Versus Wavelength λ

- Overview of the operation of a scanning sensor. Data and processing flow of the observations (Figure 6.2-2)

- How multi-channel data is built up. Intensity histograms, scatter diagrams (Figures 6.2-3 through 6.2-6)

- Logical flow of the information extraction (classification) process. Operations on a subset of the data (training fields), operations on the entire set of data (classification, presentation of results) (Figure 6.2-7)

- Concept of training fields, concept of classes. Misclassification. Measures of divergence (Figures 6.2-8, 6.2-9)

6.2.2 Model Classification Problem on the DIAL System

The purpose of the class sessions on the DIAL facility was to give each participant the opportunity to actually use the Tektronix terminal, COMTAL displays, and DIAL PMs on a representative problem.

The class was divided into two groups of two or three members each. In each group the members took turns at the Tektronix terminal with the intent that each member would go through the cycle of invoking a PM, respond to menu prompting, enter decisions and data as required, interpret Tektronix or COMTAL displays, communicate with the image and field/class files, and proceed logically from one PM to the next to produce the classified image.

Two different model problems were chosen so that two student groups could work in parallel. One group classified a portion of the LACIE intensive site while the other group classified a portion of the ERIM image.

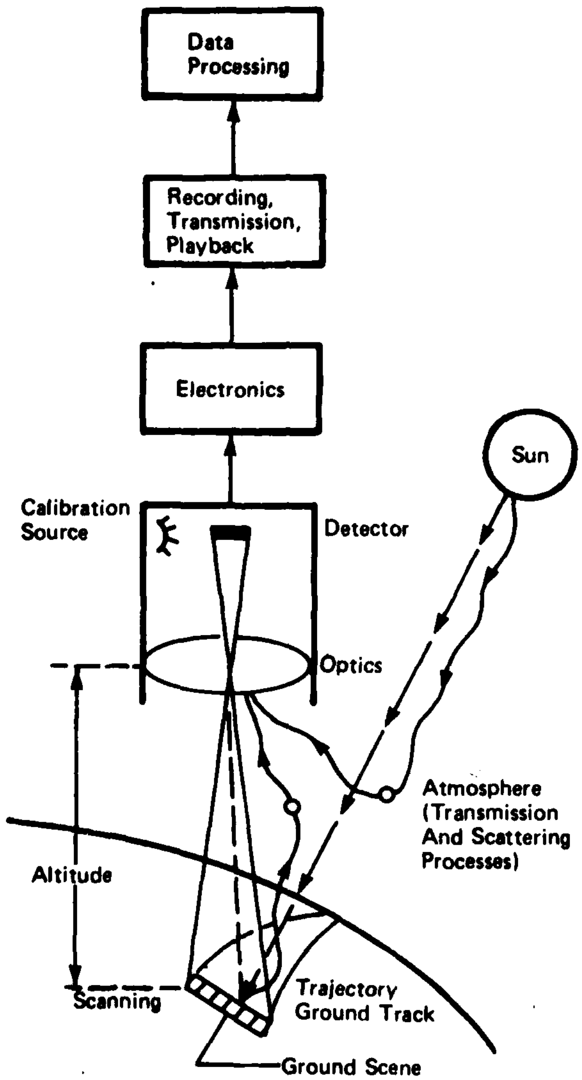


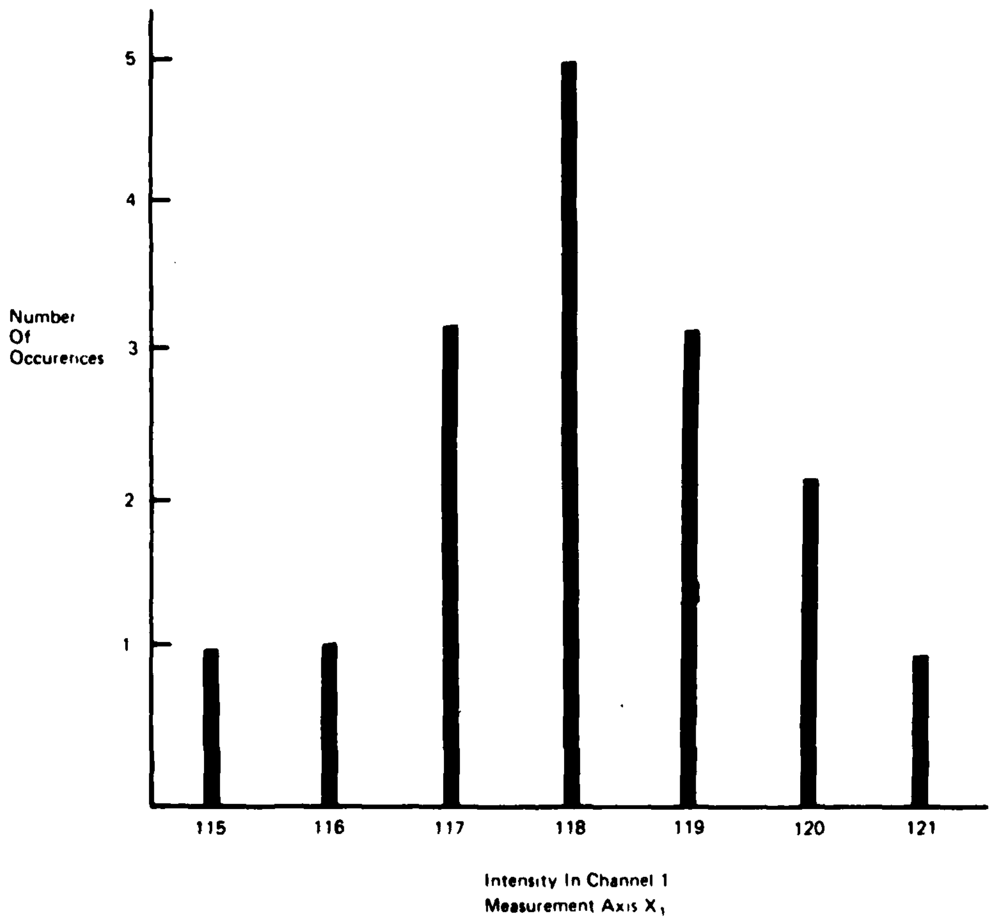
Figure 6.2-2. Operation of a Scanning Sensor

Multispectral Scanner Measurements

In Order To Illustrate The Nature And Characteristics Of A Multispectral Scanner We Consider A Scanner Which Records Observations In One Spectral Interval (Channel) For Each Ground Resolution Element.

115	117	116	118	120	118	119	119
117	118	121	120	118	119	118	117

(a) Two Lines of Typical Single Channel Data (Channel 1).



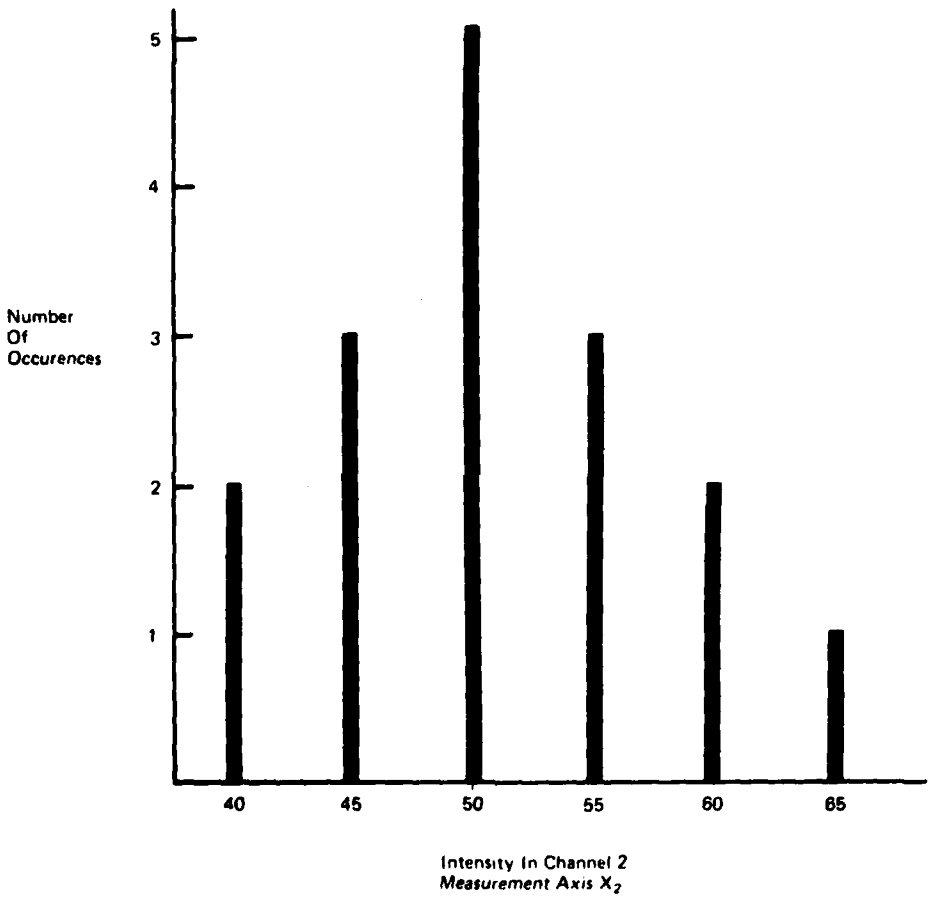
(b) Histogram of (a).

Figure 6.2-3. Multispectral Scanner Measurements - Channel 1

Consider The Case In Which A Measurement Is Taken In Each Of Two Spectral Intervals (Channels) For Each Ground Resolution Element. Let The Measurements In The Second Channel Be.

40	45	45	50	40	50	55	55
65	50	60	55	60	45	50	50

(a) Two Lines of Typical Single Channel Data (Channel 2)



(b) Histogram of (a)

Figure 6.2-4. Multispectral Scanner Measurements - Channel 2

The Two Channels Can Be Put Together By Considering That An Intensity Value Is Recorded For Each Channel For Each Ground Resolution Element.

115, 40	117, 45	116, 45	118, 50	120, 40	118, 50	119, 55	119, 55
117, 65	118, 50	121, 60	120, 55	118, 60	119, 45	118, 50	117, 50

Figure 6.2-5. Two Lines of Typical Multi-Channel (Multi-Spectral) Data - Two Channel Data.

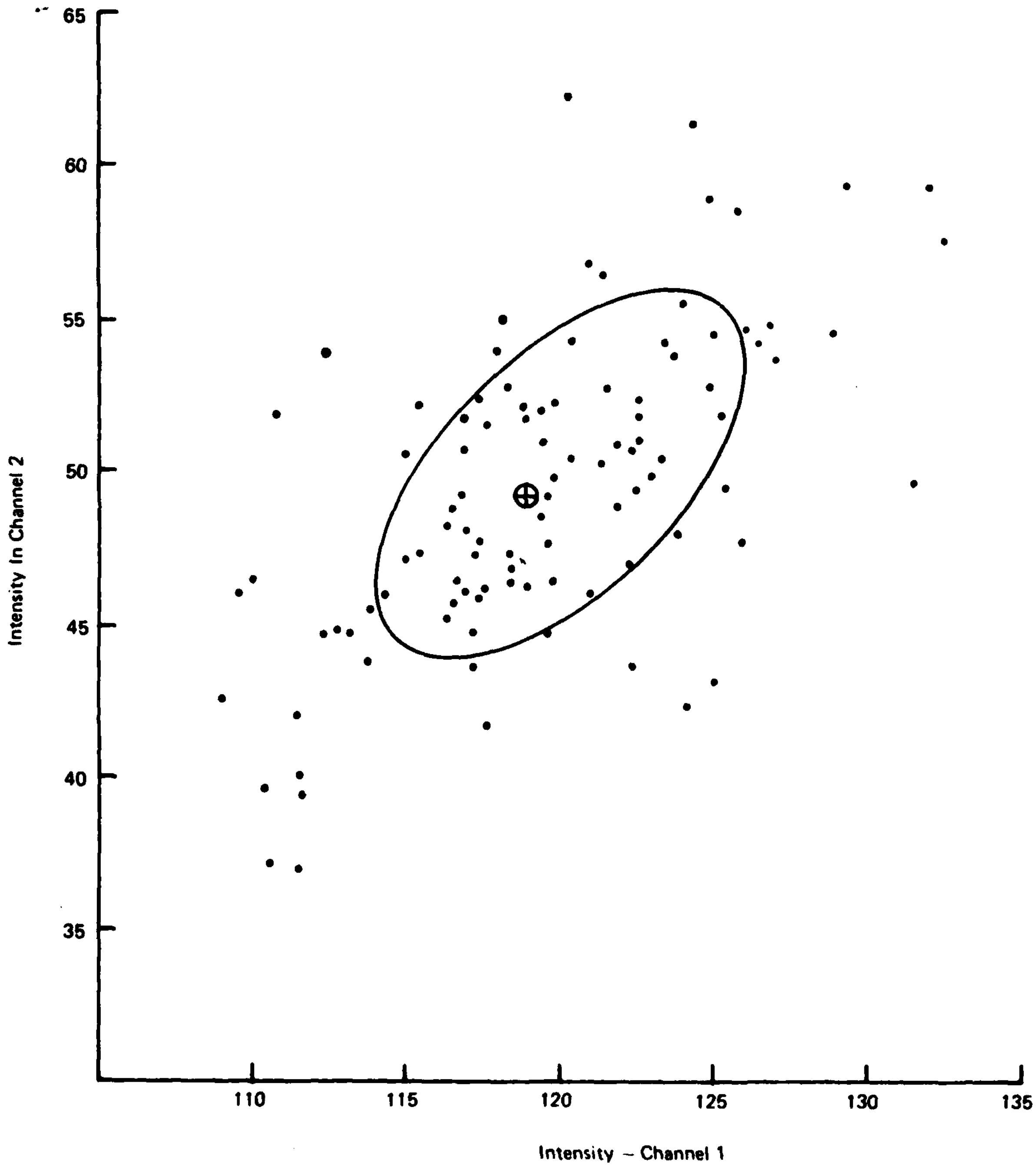


Figure 6.2-6. Scatter Diagram of Typical Multi-Channel (Multi-Spectral) Data - Two Channels

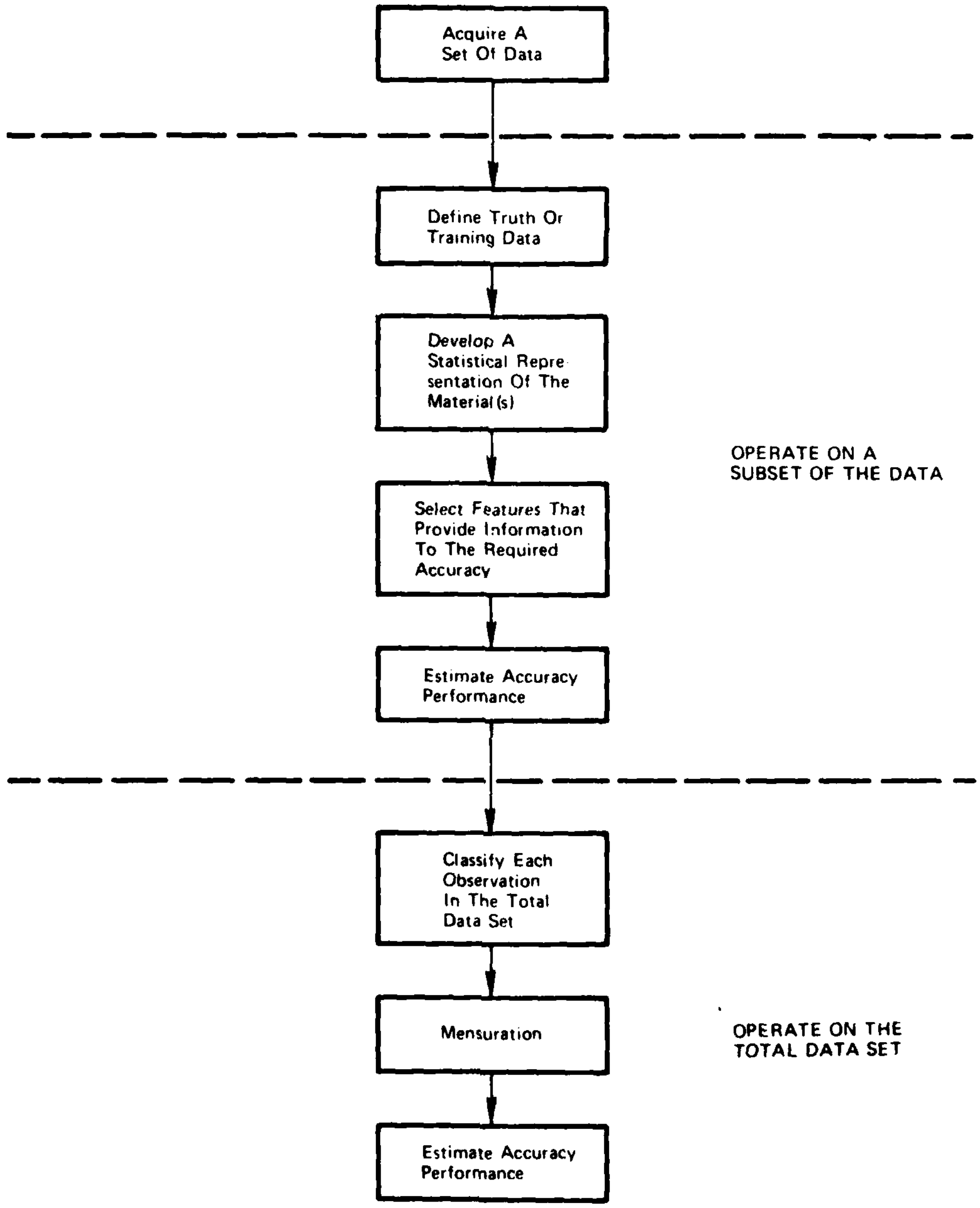


Figure 6.2-7. Logical Flow of Information Extraction (Classification) Process

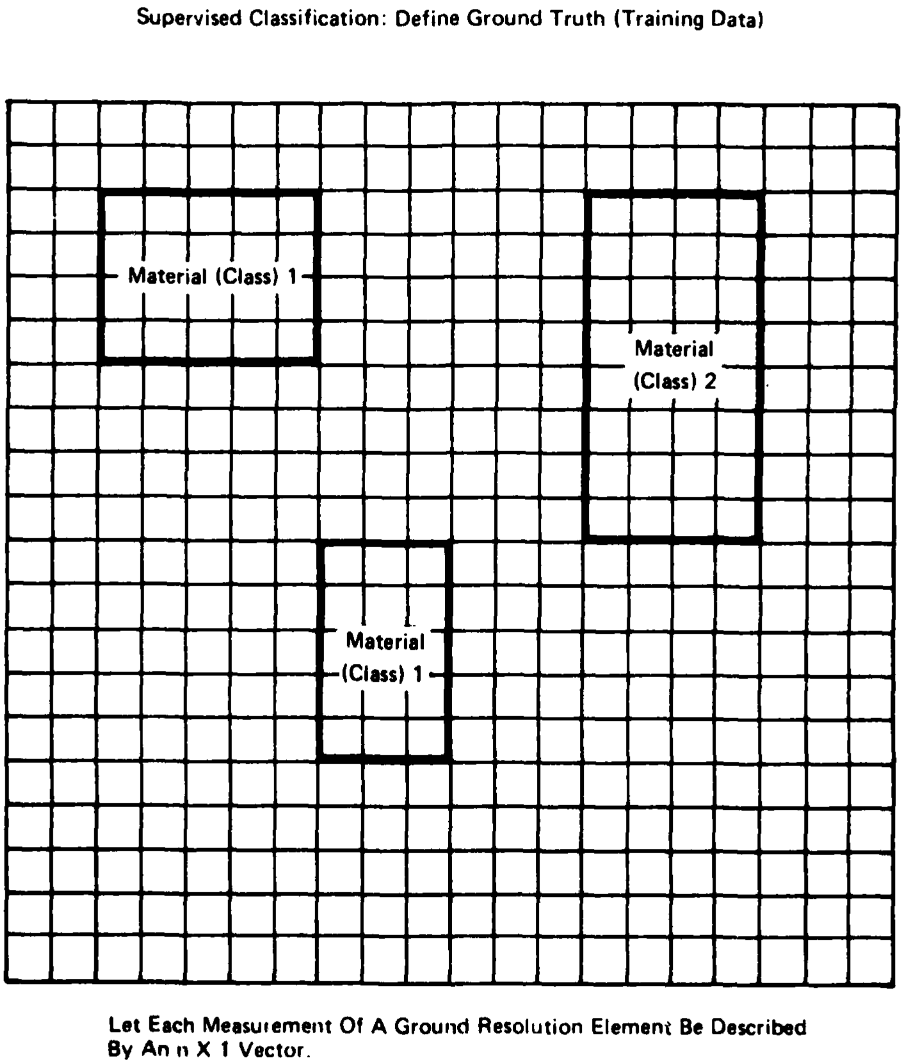
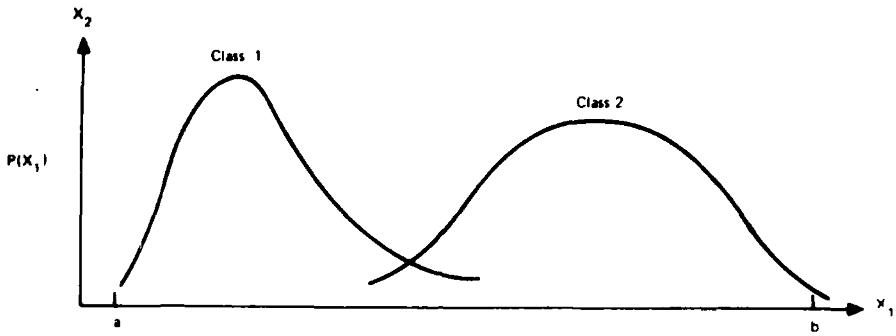
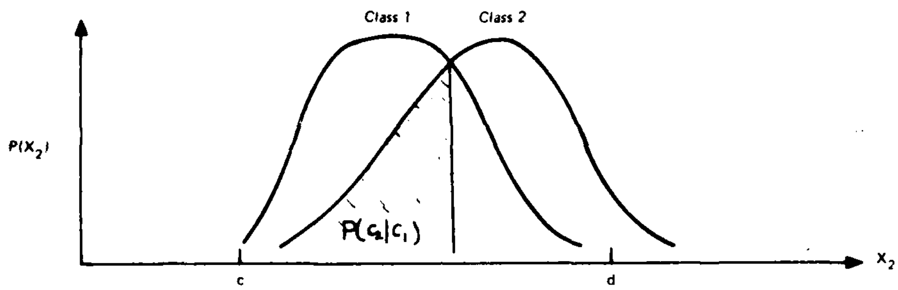


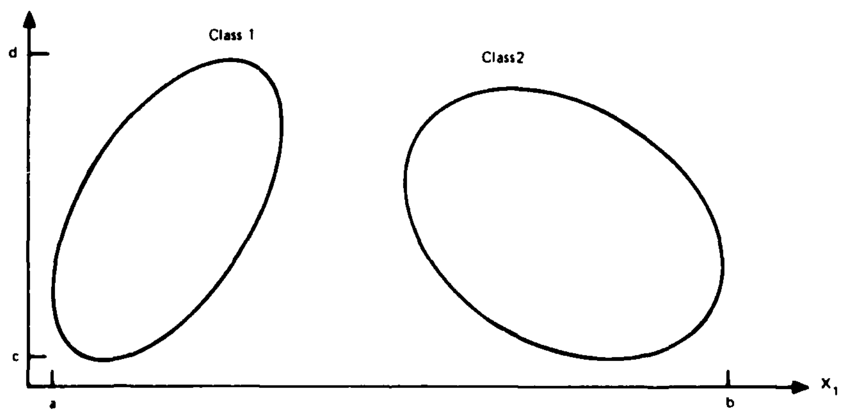
Figure 6.2-8. Typical Training Fields



(a) Well-Separated Single-Channel Classes



(b) Poorly-Separated Single-Channel Classes



(c) Well-Separated Two-Channel Classes

Figure 6.2-9. The Concept of Misclassification

Maximum Likelihood

The likelihood function of M samples whose probability density function is $p(X, \Theta_i)$ is given by

$$L(X, \Theta_i) = \prod_{r=1}^{M} P_r(X, \Theta_i) \qquad (1)$$

where Θ_i represents the statistics of the i^{th} class necessary to describe p(X). If the nx1 vector $X \varepsilon N(\mu, K)$ and a single observation is considered Eqn(1) becomes

$$L(X, \Theta_i) = \frac{1}{(2\pi)^{n/2} \left|K_i\right|^{1/2}} \exp \left\{ -\frac{1}{2} (X-\mu_i)^T K_i^{-1} (X-\mu_i) \right\} \qquad (2)$$

A sample is more likely to belong to class C_i than C_j if $L(X, C_i) > L(X, C_j)$. It is convenient to consider the log L(X), in which case

$$\log L(X, \Theta_i) = -\frac{n}{2} \log (2\pi) - \frac{1}{2} \log \left|K_i\right| - \frac{1}{2} (X-\mu_i)^T K_i^{-1} (X-\mu_i) \qquad (3)$$

Eqn(3) can be evaluated for various classes and the one for which $\log L(X, \Theta_i)$ is largest is considered to be the class of which X is an element.

Figure 6.2-10. Maximum Likelihood Classifier

In each case the area to be classified was already defined in the field/ class file and a list of classes and fields drawn up before the DIAL session began.

The classification sequence as carried out by the students proceeded as shown in table 6.2-1.

6.2.3 Comments on the Class Experience

The exercise of the unsupervised (clustering) algorithm via PM CLUSTER was not included in the curriculum since the required specification of starting parameters and the iteration-by-iteration review of the clustering process are not appropriate for the student's first contact with the sequence of classification PMs.

The number of students in a group should not exceed two, and definite assignment of classification function to students one and two should be drawn up in advance. With three or more students in a group there is a tendency for one or more of the students to observe rather than to participate.

The attempt to accomodate two groups of students simultaneously (one group at each terminal) was not entirely successful because of contention for ECS by competing PMs. The scheduled addition of further doors of ECS to the 6400 will solve that problem.

Table 6.2-1. Classification Sequence

PM	SEQUENCE OF FUNCTIONS
INTERL	1. Name the Composite Image 2. Enter Image Specifications 3. Add Images of the Respective Bands to the Composite Image 4. Build the Composite Image 5. Check Channel Correspondence
FIELDEF	6. Display Composite Images in Three Channel, False Color Form 7. Identify Area of Interest and Expand for Subsequent Field Definition 8. Define and Name Fields Using "Ground Truth" as Previously Defined by Instructors 9. Display Fields and Check That All Fields Required are in the Field/Class File. (Leave the Display Up)
CLASTAT	10. Compute Statistics for Each of the Fields Defined 11. Examine Statistics of Fields to Identify Which are Distinct as well as those Belonging to the Same Class. 12. Define Classes by Combining Fields Compute Class Signatures 13. Make Final Check of Bhattacharyya Distances Between Classes Which Will be Used in Classification

Table 6.2-1. Classification Sequence (Cont.)

PM	SEQUENCE OF FUNCTIONS
MAXLIK	14. Select Fields to be Classified (For the Model Problem, All Fields Defined in 9. are Selected)
	15. Select Classes (All Classes on Instructor's List)
	16. Select Class Parameters: Apriori Weight (Provided by Instructor), Class Character, Class Color (Student's Choice)
	17. Select Processing Options: Covariance Matrix (Fixed); Image Name and Annotation (Student's Choice)
	18. Invoke Maximum Likelihood Classification
	19. Display Class Map. Compare with Field Definition Map of 9.
	20. List Field Results and Make a Hard Copy. Review the Classification of the Training/Test Fields.
	21. Select an Appropriate Subarea of the Class Map and List, Making a Hard Copy. Examine a Few Fields Pixel-by-Pixel
	22. Threshold Result Distance Metrics at the 10% Level. Display Thresholded Class Map and Compare with Field Definition Map of 9.

Section 7

SOFTWARE DEVELOPED

One of the objectives of this contract was to develop a framework within DIAL to perform multi-channel classification and image processing experimentation. Processing modules (PM) are the elements which comprise this framework. This section presents detailed user information, including copies of the menus and requests encountered by the user, and program documentation for each PM. All software documentation assumes the user has some familiarity with DIAL. Program documentation includes a control flow given in terms of a high-level program design language (PDL), conventional descriptions of all subroutine called, and commented listings, all of which are slanted toward ease of code modification.

7.1 RATIOF PROCESSING MODULE

RATIOF computes the ratio or quotient of two single-channel images and scales the resulting image. The two images are required to have the same number of samples per line and the same number of lines; the ratioed image is the result of dividing the intensity values for each pixel in the numerator image by the intensity value of the corresponding pixel in the denominator image. The scaling process places the mean of the ratioed image at the COMTAL display mean (127.5) and includes K times (standard deviation) of the intensity distribution on each side of the mean between 0 and 255, where K is specified by the user.

Composite images made up of two or three ratioed images with the appropriate color section for each of the channels have proven to be extremely useful in the visual identification of geological structures in Landsat and other remotely-sensed images.

7.1.1 User Information

The first menu the user sees (Fig. 7.1-1) asks him to specify the numerator and denominator images. As usual, the user may type in image names if he knows them, or hit (CR) in which case his pack directory will come up on the Tektronix screen and he can specify images by number. The next menu (Fig. 7.1-2) asks the user to name the ratioed image and to enter annotation data (if desired). The final menu before processing (Fig. 7.1-3) allows the user to specify the value of K if he wants to use a value other than the default value (K=3).

The computation proceeds in two passes. On the first pass statistics are gathered for the calculation of the scaling coefficients. On the second pass, the scaled ratioed image is produced. In the course of each pass, "N percent of the computation completed" messages are written on the Tektronix screen to keep the user informed about the progress of the computation (Fig. 7.1-4).

If the two images to be ratioed chosen by the user are not compatible, that is they do not have the same number of samples per line or the same number of lines, an error message to that effect is written on

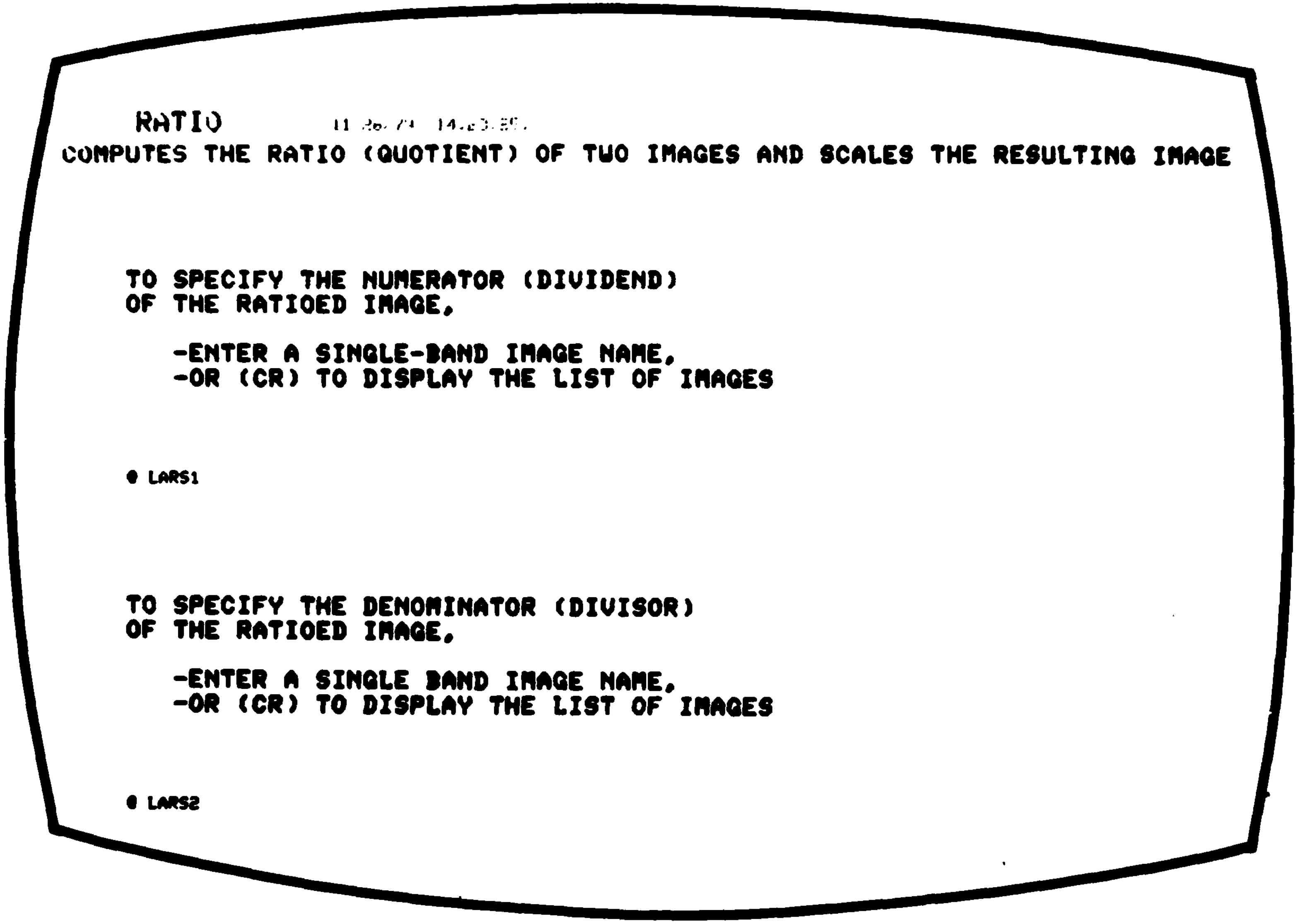


Figure 7.1-1. Menu for Names of Images

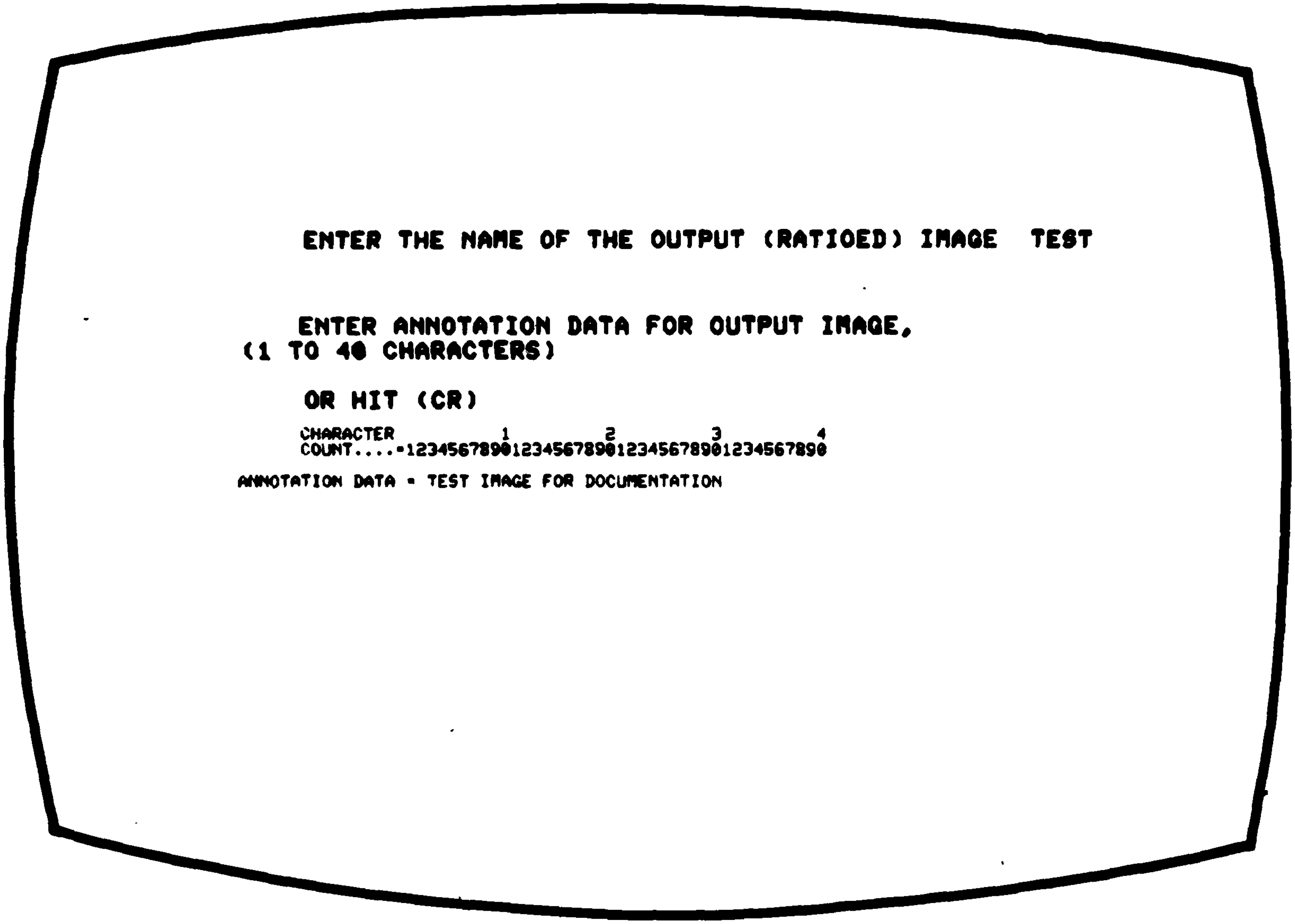


Figure 7.1-2. Menu for Output Image

```
THE RATIOED IMAGE IS TRANSFORMED SO THAT ITS
MEAN IS AT THE DISPLAY MEAN (127.5) AND
K X ITS ST. DEV. ON BOTH SIDES OF THE MEAN
COVERS THE RANGE OF DISPLAY INTENSITIES (0-255).

  -ENTER A F.P. VALUE FOR K
  -OR (CR) FOR DEFAULT VALUE K=3.0 (RECOMMENDED).  3
```

Figure 7.1-3. Menu for Scaling Factor

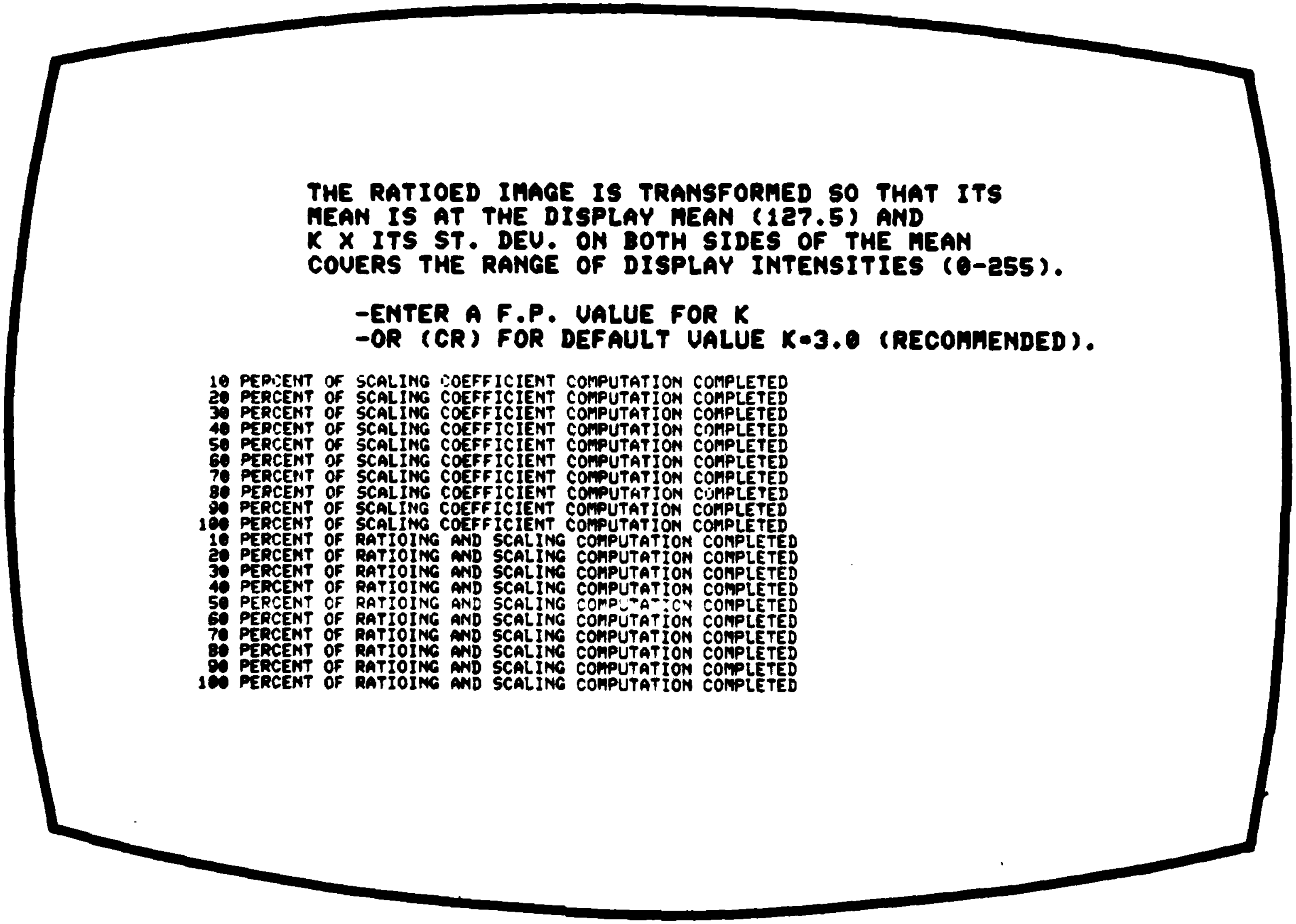


Figure 7.1-4. Status Message Display

the Tektronix screen (Fig. 7.1-5), and the user has the option of choosing another pair of images or exiting the PM.

7.1.2 RATIOF Control Flow

The control flow of RATIOF is described by the following Program Design Language (PDL):

```
RATIOF:BGNSEGMENT(MAIN)
CALL LOCATE TO SPECIFY NUMERATOR IMAGE
CALL LOCATE TO SPECIFY DENOMINATOR IMAGE
IF DIMENSIONS OF NUMERATOR NOT EQUAL TO DENOMINATOR
THEN
  IF EXIT OPTION
    EXIT PM
  ELSE
    CHOOSE NEW PAIR OF IMAGES
  ENDIF
ELSE CALL KREATE TO TO CREATE RATIOED IMAGE FILE
ENDIF
CALL LBLWRT TO WRITE RATIOED IMAGE LABEL
IF MORE THAN ONE IMAGE LINE FITS INTO ECS
THEN
  DO UNTIL NO MORE IMAGE LINES TO BE PROCESSED
    DO UNTIL NO MORE NUMERATOR LINES TO BE READ INTO ECS
      CALL DREAD TO READ PACKED IMAGE LINE INTO BUFFER
      CALL WRITEC TO WRITE PACKED IMAGE LINE INTO ECS
    ENDDO
    DO UNTIL NO MORE DENOMINATOR LINES TO BE READ INTO ECS
      CALL DREAD TO READ PACKED IMAGE LINE INTO BUFFER
      CALL WRITEC TO WRITE PACKED IMAGE LINE INTO ECS
    ENDDO
    DO UNTIL NO MORE LINES IN ECS TO BE PROCESSED
      CALL READEC TO READ PACKED NUMERATOR LINE INTO BUFFER
      CALL UNPKI TO UNPACK NUMERATOR DATA
      CALL READEC TO READ PACKED DENOMINATOR LINE INTO BUFFER
      CALL UNPKI TO UNPACK DENOMINATOR LINE
      DO UNTIL NO MORE PIXELS IN LINE TO BE RATIOED
         ACCUMULATE SUM, SUM OF SQUARES
         ISSUE PERCENT COMPLETED MESSAGE
      ENDDO
    ENDDO
```

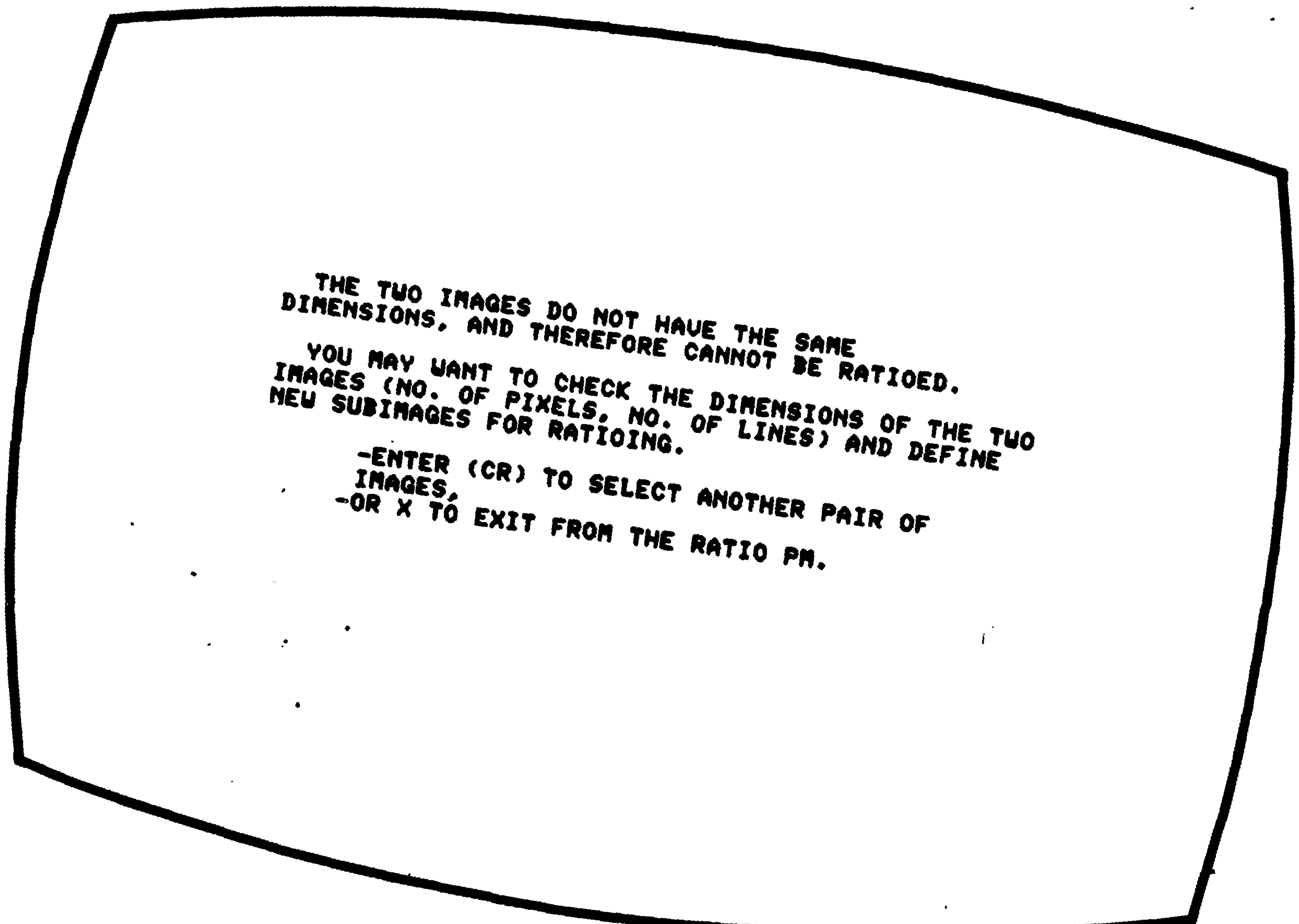


Figure 7.1-5. Error Display

```
  ENDDO
  COMPUTE SCALING COEFFICIENTS A AND B
  DO UNTIL NO MORE IMAGE LINES TO BE PROCESSED
    DO UNTIL NO MORE NUMERATOR LINES TO BE READ INTO ECS
      CALL DREAD TO READ PACKED IMAGE LINE INTO BUFFER
      CALL WRITEC TO WRITE PACKED IMAGE LINE INTO ECS
    ENDDO
    DO UNTIL NO MORE LINES IN ECS TO BE PROCESSED
      CALL READEC TO READ PACKED NUMERATOR LINE INTO BUFFER
      CALL UNPKI TO UNPACK NUMERATOR LINE
      CALL READEC TO READ PACKED DENOMINATOR LINE INTO BUFFER
      CALL UNPKI TO UNPACK DENOMINATOR LINE
      DO UNTIL NO MORE PIXELS IN LINE TO BE RATIOED AND SCALED
        RATIO
        SCALE (RADIOMETRICALLY ENHANCE)
        MINIMUM/MAXIMUM RATIOED IMAGE INTENSITY
        8-BIT RATIOED IMAGE HISTOGRAM
      ENDDO
      CALL PACKI TO PACK RATIOED IMAGE LINE
      CALL WRITEC TO WRITE PACKED LINE INTO ECS
    ENDDO
    DO UNTIL NO MORE PACKED RATIOED IMAGE LINES IN ECS TO BE
     WRITTEN TO DISK
      CALL READEC TO WRITE PACKED RATIOED IMAGE LINE INTO ECS
      CALL DWRITE TO WRITE PACKED RATIOED IMAGE LINE TO DISK
      ISSUE PERCENT COMPLETED MESSAGE
    ENDDO
    CALL DCLOSE TO CLOSE RATIOED IMAGE FILE
  ENDDO
ELSE
  ISSUE MESSAGE THAT IMAGE LINE IS TOO LARGE TO FIT INTO ECS
ENDIF
ENDSEGMENT
```

7.1.3 RATIOF Program Description

The output of RATIOF, the ratioed image, is the image produced by dividing the intensity values of the pixels in the numerator image by the intensity values of the corresponding pixels in the denominator image, and scaling the result. The numerator and denominator images are required to have the same number of samples per line and the same number of image lines; the ratioed image will of course have the same dimensions. The numerator and denominator images are not required to have the same number of bits per pixel, however.

The image formed by this decision process is then scaled by a linear transformation so that the mean of the resulting ratioed image is equal to the display mean (127.5), and K X (standard deviation) of the intensity values will be between 0 and 255, the intensity range of the COMTAL display. If $DN_i^{(N)}$ and $DN_i^{(D)}$ are the (digital) intensity values of the i-th pixels of the numerator and denominator image respectively, then the intensity of the i-th pixel of the ratioed image is:

$$DN_i^{(R)} = A * (DN_i^{(N)} / DN_i^{(D)}) + B \qquad (1)$$

the scaling requirements imply that

$$E\ (DN_i^{(R)}) = A * E\ (DN_i^{(N)} / DN_i^{(D)}) + B = 127.5 \qquad (2)$$

where E(.) is the expected value, and that

$$K \sqrt{var\ (DN_i^{(R)})} = K\ A \sqrt{var\ (DN_i^{(N)} / DN_i^{(D)})} = 127.5 \qquad (3)$$

where var(.) is the variance. Equations (2) and (3) define the scaling coefficients A and B. In the case in which the numerator and denominator intensities are Gaussian (normal) a closed form solution for the distribution of the ratioed image intensities is available, and it is theoretically possible to compute A and B from the statistics of the input images. However the computation is complicated, and the normality assumption questionable, therefore RATIOF proceeds in two passes through the data. In the first pass $E(DN_i^{(N)}/DN_i^{(D)})$ and var $(DN_i^{(N)}/ DN_i^{(D)})$ are accumulated and then used to compute A and B,

AD-A087 518 IBM FEDERAL SYSTEMS DIV GAITHERSBURG MD F/G 20/6

INTERACTIVE DIGITAL IMAGE PROCESSING INVESTIGATION. PHASE II.(U)

APR 80 W C RICE, J S SHIPMAN, R J SPIELER DAAK70-77-C-0166

UNCLASSIFIED ETL-0221 NL

2 OF 3

AD A087518

while in this second pass the values of $DN_i^{(R)}$ as given by equation (1) are computed.

The value of K is subject to a certain amount of experimentation. The default value K=3 worked well for the images on which RATIOF was tested, producing ratioed images whose histograms are roughly Gaussian. Values of K which are too small produce severely skewed histograms whose appearance on the COMTAL is less satisfactory for visual classification that the K=3 images.

In the event that $DN_i^{(D)} = 0$, which would result in overflow, $DN_i^{(D)}$ is arbitrarily set equal to 1. Some other choice might have been made, for example setting $DN_i^{(R)}$ equal to 255, but the choice is of little practical consequence, since actual image data rarely has intensity values exactly zero.

7.2 MSSRFT PROCESSING MODULE

MSSRFT is a module which accepts the LANDSAT 1 and 2 computer compatible tape (CCT) data format and reformats the data into separate band data sets for use under DIAL, directly. MSSRFT operates in batch mode, with pause messages to alert the computer operator to change input Landsat tapes if necessary. Extended core storage (ECS) is used to reduce the significant amount of processing time required to reformat these large data sets.

7.2.1 User Information

The user must supply the Landsat tape(s), and the execution card deck to the operator to use MSSRFT. For a typical execution card deck see

```
ETRR,NT1,EC160,IO4000,T4000.
TASK(TNET71355,PWSPILR,TRTS)RJSPIELER
MOUNT,VSN=PK0065,SN=LANDSAT.
REQUEST TAPE7,PE,L,NS,SV,NORING,NR,VSN=E503.
REQUEST TAPE1,AY,SN=LANDSAT,VSN=PK0065.
REQUEST TAPE2,AY,SN=LANDSAT,VSN=PK0065.
REQUEST TAPE3,AY,SN=LANDSAT,VSN=PK0065.
REQUEST TAPE4,AY,SN=LANDSAT,VSN=PK0065.
ATTACH(MSSRFT,ID=ET71355,MR=1)
MSSRFT.
CLIST(T=U)
EXIT(S)
CLIST(T=U)
789 CARD
LANDSAT     ET76969     PK0065     1
FULDAWEST1
FULDAWEST2
FULDAWEST3
FULDAWEST4
789 CARD
6789 CARD
```

Figure 7.2-1. Typical MSSRFT Execution Card Deck

Figure 7.2-1. One Landsat scene (all 4 bands) occupies approximately 35% of one CDC disk pack, and since a temporary set of data sets are created on disk along with the final or permanent set, if both temporary and final data sets are created on the same disk pack at least 70% of the pack must be free. However, the temporary data sets can be assigned to a different disk pack than the one used for the permanent data sets. Tapes must be mounted one at a time since only a single 9-track tape drive is available. Tapes will be rewound by the program and must be manually unloaded. If the CCT format is 2 or 4 input tapes for the entire scene, then the tapes must be mounted in the sequence 1 of ..., 2 of .., etc., as shown on the tape paper label.

The relatively simple execution deck is shown in Figure 7.2-1. The elements of the deck shown with underline are those parts that are variable, and the balance shown without underline should not be changed.

Deck Explanation By Line:

<u>Line</u>

1 - The job card; the job name may be changed, but the request for 9-Track tape drive, ECS, ID and execution time must remain.

2 - A standard task card.

3 - A request to mount the required pack.

4 - Request Tape 7 - this card references the input tape(s). PE (1600 bpi) may also be HD (800 bpi). Change VSN to alert operator to tape name.

5-8 - Request Tape 1 thru 4 - these cards reference the temporary data sets created during reformatting. The pack location is specified by set name and VSN, and may be on any pack or packs the user dictates. Any packs referenced should have a corresponding mount card. These data sets are automatically deleted at the end of the job.

9-10 - Attach and execute MSSRFT.

11-13 - CLIST (classified listing) feature required at ETL.

14 - Card to specify disk SN and VSN, data set ID and the number of input tapes.
Col. (1-10) - disk set name (left justified)
Col. (11-20) - data set ID (left justified)
Col. (21-30) - disk VSN (left justified)
Col. (31) - Number of input tapes (1, 2, or 4)

15 - Band 4 Dial data set name

16 - Band 5 Dial data set name

17 - Band 6 Dial data set name

18 - Band 7 Dial data set name

All of the one or more different pack(s) referenced in lines 5 thru 8, and line 14, should have a corresponding mount card.

7.2.2 MSSRFT Control Flow

READ Control Parameters
CALL Q9START to Initiate DIAL Batch-Interface

```
Allocate DIAL Output Files
CALL MSS14 to Deinterleave Left Quarter of CCT Data
IF Number of Tapes is 4
THEN PAUSE and REWIND tape
ENDIF
CALL MSS14 to Deinterleave Next Quarter of CCT Data
IF Number of Tapes is 2 or 4
THEN PAUSE and REWIND tape
ENDIF
CALL MSS14 to Deinterleave Next Quarter of CCT Data
IF NUMBER of Tapes is 4
THEN PAUSE and REWIND tape
ENDIF
CALL MSS14 to Deinterleave Right Quarter of CCT Data
Calculate processing constants
DO UNTIL EOF
    Fill ECS with a batch from each deinterleaved input data set
    IF there are batches of data left
    THEN assemble quarter pieces and store back into ECS
          Copy assembled lines from ECS to 4 DIAL data sets
ENDIF
ENDDO
CLOSE DIAL FILES
```

7.2.3 MSSRFT Program Description

Figure 7.2-2 shows pictorially the MSS format as received on the computer compatible tape (CCT). Each data set contains all lines and all bands of 1/4 width of the scene. Each record contains all the pixels and bands of 1/4 of that corresponding line. The pixels are interleaved in a 2-pixel per band, 8-pixel group. To produce individual band images

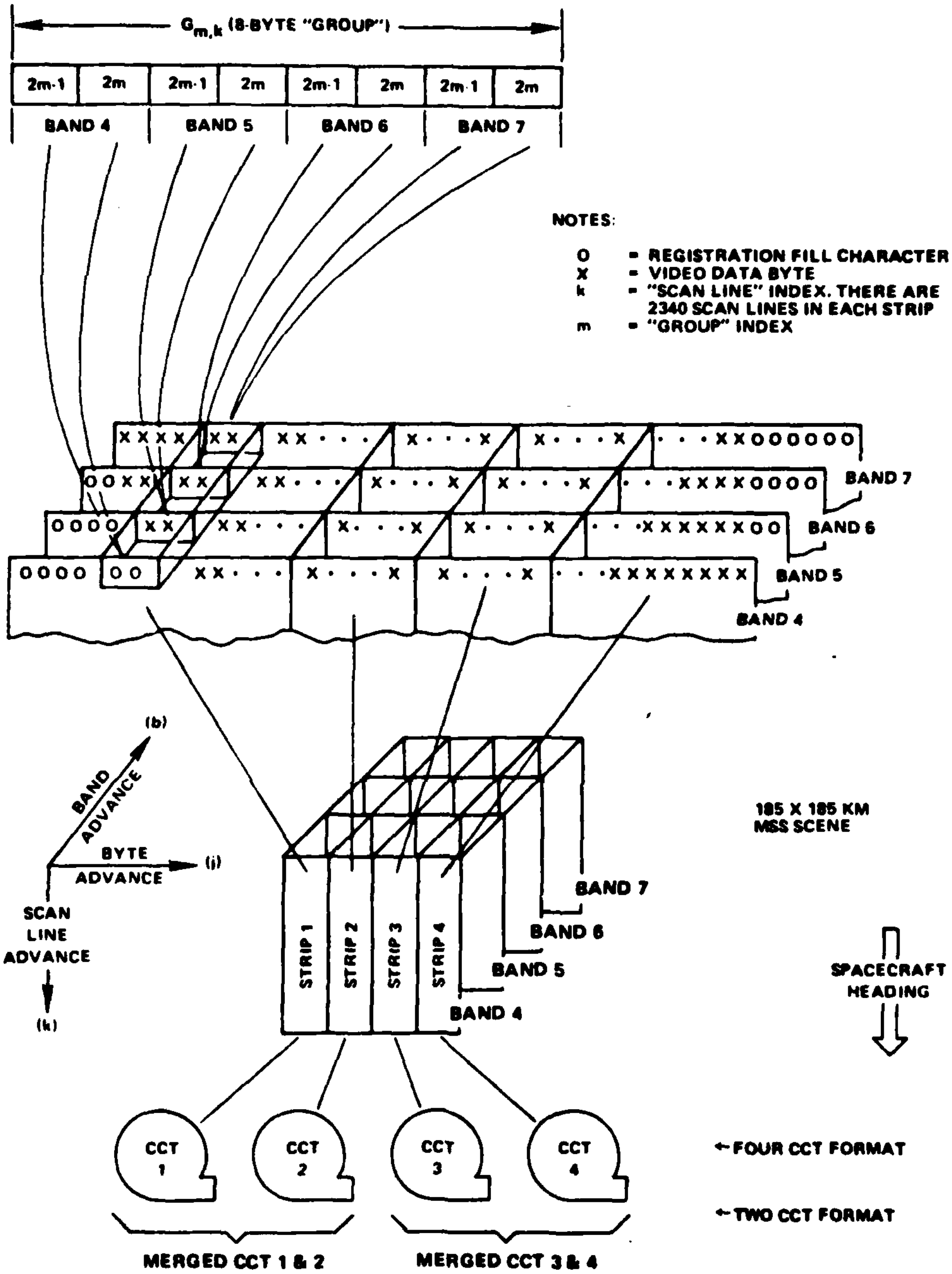


Figure 7.2-2. Bulk MSS Four-Band Scene to Interleaved CCT Format

the individual pixel pairs must be deinterleaved, and the 1/4 line segments must be reconnected.

MSSRFT operates in two steps or phases to produce deinterleaved Landsat DIAL data sets. Phase one includes reading the 4 data sets, deinterleaving each input record into individual band 1/4-lines, and writing the 1/4-line-segments to 4 temporary disk data sets. Phase two retrieves the corresponding 1/4-line-segments, connects them, and writes the complete line to the permanent data set of a particular band.

MSSRFT begins by reading the disk pack designation where the DIAL files are to be placed, and the names of the DIAL Files. The format of the names are checked and the DIAL files are started. Subroutine MSS14 is called 4 times to deinterleave each of the 4 input data sets, with a corresponding temporary disk data set created. The input data sets may be on 1, 2, or 4 CCTs. This ends phase one. Phase two consists of a major loop with three subloops. The first subloop fills ECS with the same number of records from each of the 4 disk data sets. The second subloop extracts the 4 parts of an individual line from ECS, connects the 4 into one whole line, and replaces it back to ECS. The third subloop extracts the whole lines from ECS and writes them to the corresponding band DIAL data sets. The major phase two loop continues until the temporary data sets are exhausted.

Since ECS is used to reduce execution time, there may be some interference with DIAL terminal processing if ECS is also being used by the terminal users.

Subroutine MSS14

Title - Deinterleaves 1/4 of LANDSAT image (one input data set), and outputs a "Buffer Out" file. Input format is band interleaved by double pixel, and output format is band interleaved by line (1/4 width).

Parameters

CALL MSS14 (IUI, IUO, LSBUF, LSECS, IESZ, NOBR, LSBUFA)

- IUI - Input Fortran unit number
- IUO - Output Fortran unit number
- LSBUF - Start location of core buffer
- LSECS - Start location of ECS buffer
- IESZ - Size of buffer in ECS
- NOBR - Number of bytes for this segment input record (returned).
- LSBUFA - Start of core buffer for annotation data (annotation data returned)

Description

Subroutine MSS14 performs the phase one processing on one input MSS tape. This includes reading the ID record and the annotation record (the first two records on the tape), and the major processing loop.

The major loop contains a call to fill ECS with data from the input tape, three nested subloops to deinterleave the data, and a call to copy the data from ECS to disks. The first nested subloop contains a call to copy a line from ECS and unpack it in core, followed by the second subloop. The second subloop deinterleaves an entire 1/4 line a band at a time. It includes the third subloop and the call to pack each band-1/4-line and copy to ECS. The third subloop deinterleaves one band of the 1/4-line by copying a pixel pair from each group of 8 to a buffer assembly area.

Subroutine FEUTLPU

Title - Transfers data from ECS, and unpacks data using UNPKI or UNPKF

Parameters

CALL FEUTLPU (IUPI, LSECS, NPIX, NBITS, NCH, IPST, LSBUF, IBSZ,
NPR, LPUNDR)

IUPI - Unpack type indicator, 0 - UNPKI
1 - UNPKF

LSPECS - Start location of line in ECS

NPIX - Number of super pixels in line

NBITS - Number of bits per subpixel

NCH - Number of channels

IPST - Starting superpixel to be unpacked

LSBUF - Start of buffer in core

IBSZ - Size of buffer in core

NPR - Number of superpixels unpacked (returned)

LPUNPR - Last pixel unpacked (returned)

Description

Subroutine FEUTLPU calculates the maximum number of superpixels (in multi-channel data, a superpixel is the group of a corresponding pixel from each channel) that can be unpacked in the available core buffer size, taking into account the number of unpacked words and the length of the packed line in the core buffer. The unpacked data (using either UNPKI or UNPKF is placed in the beginning of the buffer.

Subroutine FUETLDE

Title - Transfers data from a FORTRAN unit or a DIAL file to ECS.
Lines are stacked in sequence in ECS.

Parameters

CALL FEUTLDE (IUI, FNAME, LSRT, MLNS, NPIX, NBITS, NCH, LSBUF, LSECS, IESZ, NLECSR, NWR, IEOFR)

- IUI - 0 - Indicates DIAL file (IEOFR is not returned) (FNAME, LSRT, MLNS, NPIX, NBITS, and NCH are unused)
- FNAME - DIAL input file number
- LSRT - Start line in DIAL file
- MLNS - Maximum number of lines to be transferred
- NPIX - Number of superpixels in line
- NBITS - Number of bits per subpixel
- NCH - Number of channels
- LSBUF - Start location of buffer in core
- LSECS - Start location of buffer in ECS
- IESZ - Size of buffer in ECS
- NLECSR - Number of lines transferred to ECS (returned)
- NWR - Number of words in packed line (returned)
- IEOFR - End of file indicator (returned)
  - 0 - Not EOF
  - 1 - EOF

Description

Subroutine FEUTLDE determines the length of a packed line, calculates the number of lines that will fit into the size buffer in ECS, and then copies that number of lines into contiguous ECS.

Subroutine FEUTLED

Title - Transfers data from ECS to a FORTRAN unit or a DIAL file.
If there is more than one line in sequence in ECS, then
FEUTLED assumes they are contiguous.

Parameters

CALL FEUTLED (IOU, FNAME, NPIX, NBITS, NCH, LSBUF, LSECS)

IOU - 0 - Indicates DIAL file
N > 0 - specifies output FORTRAN unit, FNAME is not used.
FNAME - DIAL output file name
NLNS - Number of lines to be transferred
NPIX - Number of superpixels in line
NBITS - Number of bits per subpixel
NCH - Number of channels
LSBUF - Start location of buffer in core
LSECS - Start location of data in ECS

Description

Subroutine FEUTLED calculates the size of a pack line, then loops the number of lines between copying data from ECS and writing to disk.

Subroutine FEUTLUP

Title - Packs data using PACKI or PACKF and transfers data to ECS.

Parameters

CALL FEUTLUP (IUPI, LSECS, NBITS, NCH, IPST, LSBUF, IBSZ, NP,
LSBUFW)

IUPI - Pack type indicator, 0 - PACKI
1 - PACKF

LSECS - Start location of line in ECS

NBITS - Number of bits per subpixel

NCH - Number of channels

IPST - Starting superpixel to be packed

LSBUF - Start of buffer in core, and start of unpacked data

IBSZ - Size of buffer in core

NP - Number of superpixels to be packed

LSBUFW - 61 word work area; LSBUFW(1) must equal 0 at the beginning of a line to be packed. If more than one call is required to process the entire line, LSBUFW stores any pixels needed to complete the next whole word.

Description

Subroutine FEUTLUP is designed as a complement to FEUTLPU, that is, it will pack part of lines that are to be packed and retain any odd subpixels necessary to complete integer word(s) when it is next invoked. Of course, if an entire line is to be packed at once, it will do that also.

FLEUTLUP packs data in four possible steps each time it is invoked. Which steps are selected depends on the conditions upon entry to FEUTLUP.

Step 1 is selected if there is data in LSBUFW that was left over from a previous incomplete line PACK. If it is selected, the number of words needed to complete integer packed word(s) are copied from the start of LSBUF to LSBUFW and those integer packed word(s) are written to ECS.

Step 2 is selected if, after step 1, there are enough unpacked subpixels left to produce integer packed words. If it is selected, as many subpixels (unpacked, one per word) as possible that will produce integer packed words are packed and written to ECS. This will be done in more than one loop if there is insufficient buffer space to pack the entire amount at one time.

Step 3 is selected if, after step 1 and 2, there are subpixels remaining in the main buffer. If it is selected, the remaining subpixels are copied to LSBUFW.

Step 4 is selected if, after steps 1, 2 and 3 there are subpixels remaining in LSBUFW. If it is selected, the amount of additional unpacked words necessary to produce integer packed words are zeroed and the result is packed and written to ECS. If this is not the end of the line, this area in ECS will be over written the next time FEUTLUP is invoked.

7.3 KLTRAN PROCESSING MODULE

KLTRAN provides the capability to generate uncorrelated principal components of a multi-dimensional (composite) image by means of the Karhunen-Loève (K-L) transformation. The K-L transformation is also known as the principal component transformation, the eigenvector transformation, or the Hotelling transformation. This unitary transformation is based upon the eigenvectors of the image which are computed and displayed for the user's evaluation. While obtaining the principal components the PM allows the user to radiometrically enhance them, without requiring additional computation.

7.3.1 User Information

This processing module is invoked by entering KLTRAN on the Tektronix. The user is then requested to specify the name of a field/class data file or to enter a (CR) for a display of all the field/class data files on the user's disk pack. In order to develop the K-L transformation, the mean vector and covariance matrix of the entire image or some specified area within the image are needed. CLASTAT is used for this purpose and the next request of the user is for the class name, which has been assigned to the area statistics, so that they can be retrieved and utilized in establishing the K-L transformation.

From this point on all processing is directly controlled by the menu presented in Figure 7.3-1.

7.3.1.1 Option 1 Statistical Parameters

Statistical parameters for the class selected are displayed for the user's evaluation. Figure 7.3-2 is representative of the information presented in this display. It is for the class/area WOODS which has six bands (6 dimensional data) and contained 76.14 percent of its information content in the first two eigenvalues/principal components. Column six is the angle in degrees between the mean vector and the eigenvector associated with the eigenvalue in that row. All other information is self-explanatory. Also a four option menu is displayed to give the user additional statistical information about the area or to return to the main KLTRAN processing menu. Figures 7.3-3 to 7.3-5 give examples of the other statistical displays.

<u>Available KLTRAN Processing Options</u>

1. List Statistical Parameters of Original Bands
2. Select Image (Composite) to be Transformed
3. Perform K-L Transformation of Image
4. Display Composite Images
5. Deinterleave Transformed Image (Composite)
6. Terminate KLTRAN Processing

Figure 7.3-1. KLTRAN Processing Options Menu

STATISTICAL PARAMETERS--ORIGINAL BANDS--AREA
WOODS

BAND	MEAN	VARIANCE	EIGEN VALUES	FRACTION OF TOTAL VAR.	ANGLE MEANXEIGEN.
1	93.452	19.408	188.94	.45590	36.732
2	49.267	62.169	126.63	.30553	66.356
3	41.179	36.476	45.194	.10905	83.823
4	97.705	175.12	25.290	.61022E-01	76.763
5	83.215	97.793	15.601	.37644E-01	74.227
6	59.078	23.471	12.785	.30848E-01	75.950

AVAILABLE OPTIONS ARE

1 TEK. DISPLAY OF COVARIANCE MATRIX
2 TEK. DISPLAY OF CORRELATION MATRIX
3 TEK. DISPLAY OF EIGENVECTORS
4 RETURN TO KLTRAN PROCESSING
ENTER OPTION NUMBER(1-4)

Figure 7.3-2. Statistical Parameters-Original Bands

COVARIANCE MATRIX

ROW	COLUMN 1/9	2/10	3/11	4/12	5/13	6/14	7/15	8/16
1	19.488	14.227	10.963	3.5329	13.151	7.9012		
2	14.227	62.169	23.613	9.3336	35.654	13.065		
3	10.963	23.613	36.476	3.5526	20.893	12.678		
4	3.5329	9.3336	3.5526	175.18	85.041	1.2075		
5	13.151	35.654	20.893	85.041	97.783	18.414		
6	7.9012	13.065	12.678	1.2075	18.414	23.471		

AVAILABLE OPTIONS ARE

1 TEK. DISPLAY OF COVARIANCE MATRIX
2 TEK. DISPLAY OF CORRELATION MATRIX
3 TEK. DISPLAY OF EIGENVECTORS
4 RETURN TO KLTRAN PROCESSING
ENTER OPTION NUMBER(1-4) .

Figure 7.3-3. Covariance Matrix

CORRELATION MATRIX

ROW	COLUMN 1/9	2/10	3/11	4/12	5/13	6/14	7/15	8/16
1	1.0000	.40058	.41106	.60601E-01	.30187	.37080		
2	.40058	1.0000	.40606	.89458E-01	.45787	.34208		
3	.41106	.40606	1.0000	.44448E-01	.34002	.43300		
4	.60601E-01	.89458E-01	.44448E-01	1.0000	.18136	.18836E-01		
5	.30187	.45787	.34002	.18136	1.0000	.38436		
6	.37080	.34208	.43300	.18836E-01	.38436	1.0000		

AVAILABLE OPTIONS ARE

1 TEK. DISPLAY OF COVARIANCE MATRIX
2 TEK. DISPLAY OF CORRELATION MATRIX
3 TEK. DISPLAY OF EIGENVECTORS
4 RETURN TO KLTRAN PROCESSING
ENTER OPTION NUMBER(1-4)

Figure 7.3-4. Correlation Matrix

TRANSPOSE OF EIGENVECTORS

EIGENVECTOR	COMPONENTS							
	1/9	2/10	3/11	4/12	5/13	6/14	7/15	8/16
1	.76796E-01	.21052	.11787	.55649	.37980	.77986E-01		
2	.17232	.47646	.30308	-.45664	.52925	.21773		
3	.18921	.59048	.30387	.69745E-01	-.56888	.50655E-01		
4	.21734	-.50217	.61359	.21536E-01	-.44588E-01	.45981		
5	.40099	.60401E-01	-.50096	.10646E-01	-.93658E-01	.58830		
6	.84887	-.12152	-.33317E-01	-.13624E-01	.45606E-01	-.51141		

AVAILABLE OPTIONS ARE

1 TEK. DISPLAY OF COVARIANCE MATRIX
2 TEK. DISPLAY OF CORRELATION MATRIX
3 TEK. DISPLAY OF EIGENVECTORS
4 RETURN TO KLTRAN PROCESSING
ENTER OPTION NUMBER(1-4)

Figure 7.3-5. Eigenvectors

7.3.1.2 Option 2 Composite Image Selection

The user is requested to supply the name of the composite image to use in the K-L transformation or to enter a (CR) which will display a list of the names of all images on the disk. If the image is not a composite (greater than 1 band) then a message will be presented along with another request for the name of the composite image. This option needs to be selected before option 3 is exercised.

7.3.1.3 Option 3 Perform K-L Transformation

When this option is selected the first function performed is the generation of the unitary transformation matrix G. It is possible to radiometrically enhance the resulting image by modifying the normal K-L transformation matrix. Various enhancement options are available to the user.

1. No contrast enhancement.

2. Divide by $\sqrt{N}$ (N is dimension of data or number of bands) to compensate for the added range of the transformed data.

3. For each principal component (band) the pixel range will include H standard deviations on each side of the mean value. Here the ratios among the variances of the principal components are no longer maintained (H is requested of the user).

4. To preserve the ratios among the variances of the principal components the first principal component is radiometrically enhanced as in option 3, while the others are enhanced proportionally.

After selecting one of the above the actual pixel vector by pixel vector transformation is started. Since N(N+1) multiplies and additions are required per pixel, where N is the number of bands, status messages giving the percent of processing completed are displayed. Also the user is requested to name (up to 40 characters) the resulting transformed image. This option terminates with a display giving the statistical parameters of the principal components. An example is given in Figure 7.3-6. It also indicates which enhancement option was used and the name of the output image. In the example option 2 was used and WILLIAM was the name of the output image.

7.3.1.4 *Option 4 Display of Composite Images*

Selecting this option gives the user the capability to display a particular band of either the original or principal component image. The band is displayed with a magnification of 1 or less.

7.3.1.5 Option 5 Deinterleave Composite

This option gives the user the capability to deinterleave the composite and form a DIAL type image. This image can then be investigated and manipulated by any of the large repertoire of single band (one dimensional) analysis routines available to the user. The composite image is not changed by this operation and the user must save the new image before LOGOFF is entered.

STATISTICAL PARAMETERS OF PRINCIPAL COMPONENTS

ENHANCEMENT OPTION 2 IMAGE WILLIAM

BAND	MEAN	VARIANCE	EIGEN VALUES	FRACTION OF TOTAL VAR.
1	126.94	7.8418	31.724	.73741
2	126.99	2.5208	10.203	.23705
3	127.00	.16117	.64299	.15156E-01
4	127.00	.11044	.44196	.10385E-01

ENTER(CR) TO RETURN TO KLTRAN OPTIONS MENUE

Figure 7.3-6. Statistical Parameters of Principal Components

7.3.1.6 Option 6 Terminate Processing

Selecting this option causes all files to be returned and terminates KLTRAN processing.

7.3.2 KLTRAN Control Flow

The control flow of KLTRAN is described by the following PDL:

```
KLTRAN: BGNSEGMENT (MAIN)
TEKDISPLY KLTRAN TITLE MENU
CALL LOCATE TO SELECT FIELD/CLASS FILE
CALL FETEKDF TO SELECT CLASS STATISTICS FOR K-L TRANSFORMATION
CALL FACANL TO CALCULATE EIGENVALUES AND EIGENVECTORS
DO UNTIL OPTION 6 IS SELECTED
CALL TEKVAL TO SELECT PROCESSING OPTION (6 MAIN OPTIONS)
CASENTRY (LIST STATISTICAL PARAMETERS, SELECT COMPOSITE IMAGE, PERFORM
          K-L TRANSFORM, DISPLAY IMAGES, DEINTERLEAVE IMAGE, TERMINATE)
CASE 1 (LIST STATISTICAL PARAMETERS)
     CALL TEKSTAT TO TEKDISPLY AREA STATISTICS
          DO UNTIL OPTION 4 IS SELECTED
          CALL TEKVAL TO SELECT 1 OF 4 PROCESSING OPTIONS
          CASENTRY (COVARIANCE, CORRELATION, EIGENVECTORS, END)
          CASE 1 (COVARIANCE)
               CALL TEKMSG TO TEKDISPLY COVARIANCE MATRIX
          CASE 2 (CORRELATION)
               CALL TEKMSG TO TEKDISPLY CORRELATION MATRIX
          CASE 3 (EIGENVECTORS)
               CALL TEKMSG TO TEKDISPLY EIGENVECTORS
          CASE 4 (END)
               RETURN TO MAIN MENUE
          END CASE
CASE 2 (SELECT COMPOSITE IMAGE)
     CALL LOCATE TO SELECT COMPOSITE IMAGE
     VALIDATE COMPOSITE IMAGE
CASE 3 (PERFORM K-L TRANSFORMATION)
     CALL UNITARY TO CALCULATE TRANSFORMATION MATRIX AND TRANSFORM PIXELS
          CALL TEKVAL TO SELECT ENHANCEMENT OPTION (4 OPTIONS)
          DO UNTIL ONE OPTION IS SELECTED
          CASENTRY (NO ENHANCEMENT, 1/ SQRT(N), H STD DEV, H STD DEV
                    ALL PROPORTIONAL)
```

```
          CASE 1 (NO ENHANCEMENT)
               SET ALL A(I) = 1
          CASE 2 ( 1/SQRT (N) )
               SET A(I) = 1./SQRT (N) ALL I
          CASE 3 (H STD DEVS.)
               SET A(I) = D/H* SQRT (LAMDA (I) )  I=1,N
          CASE 4 (H STD DEV ALL PROPORTIONAL)
               SET A(I) = D/H*SQRT (LAMDA(1) )  I=1,N
          END CASE
          COMPUTE UNITARY TRANSFORMATION MATRIX
          CALL KREATE TO REQUEST AND CREATE OUTPUT IMAGE
          COMPUTE TRANSFORMED PIXEL INTENSITY VALUES
          CALL DCLOSE TO CLOSE NEW IMAGE
          CALCULATE STATISTICS OF NEW IMAGE
          CALL TEKWRT TO TEKDISPLAY IMAGE STATISTICS
CASE 4 (DISPLAY IMAGES)
     DO UNTIL NO MORE IMAGES ARE SELECTED
     CALL TEKRD TO SELECT IF THE ORIGINAL OR PRINCIPAL COMPONENT
                              IMAGE IS TO BE DISPLAYED
     CALL DSPCLA TO SELECT WHICH BAND IS TO BE DISPLAYED AND DISPLAY IT
CASE 5 (DEINTERLEAVE IMAGE)
     CALL TEKRD TO SELECT IF ORIGINAL OR PRINCIPAL COMPONENT IMAGE IS
                          TO BE DEINTERLEAVED
     CALL TEKVAL TO SELECT WHICH BAND IS TO BE DEINTERLEAVED
     CALL SEPAR TO CREATE THE NEW IMAGE
          CALL KREATE TO NAME AND CREATE A NEW IMAGE
CASE 6 (TERMINATE)
     CALL DRETURN TO RETURN DATA SETS TO SYSTEM
     CALL FINIS
```

7.3.3 KLTRAN Program Description

All subroutines developed for KLTRAN are described below:

7.3.3.1 Subroutine EIGEN

Title - Computes the eigenvalues and eigenvectors of a real symmetric matrix

Parameters

Call EIGEN (AA, A, E, R)

AA - Input original symmetrix matrix

N - Order of Matrix AA, A and R

A - A supplied work area for the matrix instead of destroying original matrix during the computation

E - Eigenvalues (one dimensional array)

R - Resultant matrix of eigenvectors (stored columnwise in same sequences as eigenvalues)

Remarks and Restrictions

Original matrix (AA) must be real symmetric and of order 16 or less. Matrix AA cannot be in the same location as matrix R. A dimension of 16 is assumed for array sizes.

Description

Diagonalization method originated by Jacobi and adapted by von Neumann for large computers as found in "Mathematical Methods for Digital Computers," Vol 1, edited by A. Ralston and H.S. Wilf, John Wiley and Sons, New York, 1962, Chapter 7, is used in the computation.

7.3.3.2 Subroutine FACANL

Title - Determines principal factors (eigenvalues and eigenvectors) associated with a sample covariance matrix.

Parameters

Call FACAL (NAMERC, NREC, CLASSN, COV, BUFF, SM, ND, E, R, D)

Input

NAMEFC - Name of field/class data set

NREC - Record number of class of interest

COV - Covariance matrix (16 x 16 array)

SM - Mean vector

ND - Dimension of COV

D - Scratch area of dimension (16 x 16)

BUFF - Buffer area to hold field/class record-700 words

CLASSN - Name of class associated with covariance matrix

Output

E - A ND by 4 array with

1 st column = ordered eigenvalues

2 nd column - fractional contribution of the Ith eigenvalue to total variance

3 rd column - fractional contribution of the first I eigenvalues to the total variance

4 th column = angle (deg.) between mean vector and Ith eigenvector

R - Eigenvectors stored columnwise and corresponding in order to E.

Description

This routine calls EIGEN to compute the eigenvalues and eigenvectors of the sample covariance matrix. It then orders the eigenvalues and calculates the fractional contribution each eigenvalue contributes to the total variance and the contribution of the first I eigenvalues. It also calculates the angle in degrees that each eigenvector makes with the mean vector.

7.3.3.3 Subroutine MXV

Title - Multiplication of matrix and vector plus translation

Parameters

```
CALL MXV (GBAR, X, NPIX, B, E1, E2, ND)
```

Input

GBAR - Transformation matrix (16 x 16 array)
X - Pixel intensity vectors stored columnwise ND * NPIX values
NPIX - Number of pixels in array X
ND - Vector dimension and order of GBAR
B - Translation vector

Output

X - Transformed pixel intensity vectors
E1 - Array of first moment sums, ND values
E2 - Array of second moment sums, ND values

Description

The matrix GBAR is multiplied by the vector X and the resulting vector is added to the vector B. Also first and second moment values are accumulated.

7.3.3.4 Subroutine SEPAR

Title - Creates a single band image from a composite image.

Parameters

CALL SEPAR (ND, NBIT, NAMEC, NPIX, NPIXT, NWDS, NLINES, JCHNA, ILNTH, BUFFA, BUFFB)

NB - Number of bands or dimension of composite image
NBIT - Number of bits per pixel
NAMEC - 4 word array containing name of composite image
NPIX - Number of pixels per line of imagery
NPIXT - Number of values per record of imagery (ND times NPIX)
NWDS - Number of 60 bit words per record of the composite image
NLINES - Number of lines or records in image
JCHAN - Selected band to be deinterleaved
ILNTH - Flag
= 0 if length of BUFF2 is .GE. NPIXT
= actual length of BUFF2 if length of BUFF2 .LT. NPIXT
BUFFA - User supplied buffer to hold packed line (NWDS words)
BUFFB - User supplied buffer to hold unpacked pixel values ideally NPIXT long but must be at least 60 times ND words long

Description

Requests the user to supply an image name and then creates an image where each pixel value is the JCHAN element of the composite's pixel vectors. The composite image is not modified.

7.3.3.5 Subroutine TEKSTAT

Title - Displays on the TEKTRONIX statistical parameters derived from the original bands of the image.

Parameter

```
CALL TEKSTAT (SM, COV, EIGVEC, EIGPAR, CNAME, ND, TEMP, TEMPB)
      SM - Array of sample means
     COV - Covariance matrix (16 x 16)
  EIGVEC - Eigenvectors stored columnwise
  EIGPAR - A 16 x 4 array
           1 st column = ordered eigenvalues
           2 nd column = fraction contribution of Ith eigenvalue to
                         total variance
           3 rd column = Ith value is fractional contribution of
                         first I eigenvalues to the total variance
           4 th column = Angle (deg) between mean vector and Ith
                         eigenvector
   CNAME - Class name (4 word array)
      ND - Dimensionality of data and signatures
    TEMP - Temporary storage array
   TEMP3 - Temporary storage array
```

Description

Displays the mean vector, variances, eigenvalues, fraction contribution of Ith eigenvalue to total variance and the angle between the mean vector and the Ith evigenvector on the Tektronix. Also the user has the option to display the covariance matrix, the correlation matrix and the eigenvectors.

7.3.3.6 Subroutine UNITARY

Title - Performs a unitary transformation of a composite image.

Parameters

```
CALL UNITARY (ND, NBIT, NAMEC, NPIXT, NPIX, NWDS, NLINES, SM,
              EIGVEC, EIGPAR, NAMET, GBAR, BUFF1, BUFF2, ILNTH)
```

ND - Dimension of image and transform

NBIT - Number of bits per pixel

NAMEC - 4 word array containing name of composite image

NPIXT - Number of pixels per line (ND values per pixel)

NWDS - Number of 60 bit words/packed line

NLINES - Number of lines in image

SM - Image sample mean vector

EIGVEC - Eigenvectors of image stored by column (ND of them)

EIGPAR - ND by 4 array with

- 1 st column = Ordered eigenvalues
- 2 nd column = Fraction contribution of Ith eigenvalue to total variance
- 3 rd column = Fraction contribution of first I eigenvalues to total variance

4 th column = Angle (deg) between eigenvector and mean vector

NAMET - Name of resulting transformed image 4 word array

GBAR - Buffer to hold transformation matrix (NDxND) must be 256 words

BUFF1 - Buffer to hold packed line (NWDS words)

BUFF2 - Buffer to hold unpacked pixel values (must be at least 60*ND words long)

ILNTH - Flag

= 0 if length of BUFF2 is .GE. NPIXT

= Length of BUFF2 if length of BUFF2 is .LT. NPIXT

NPIX - Number of pixels/line

Description

This routine first requests of the user to select one of four radiometric enhancement options:

1. No enhancement

2. Scale by 1/SQRT(ND) A(I) = 1/SQRT(ND)

3. Include H standard deviations on each side of the mean value
   A(I) = D/H*SQRT(EIGENVALUE(I)

4. Include H standard deviations on each side of the mean
   A(I)=D/H *SQRT(EIGENVALUE(1))
   where D is 1/2 the dynamic range of the pixel size.

After selecting one of the above options the linear scaling coefficients A and B are calculated with the prime objective of spreading the information within the range of 0 to $2^{(NBIT)}-1$. The calculation for A is given above and

$$B(I) = \mu - A(I)* \sum_{J=1}^{ND} G(I,J)SM(J)$$

where

μ = center of pixel range
G(I,J) = matrix of eigenvectors
SM(J) = sample mean vector

Next the vector A(I) is incorporated into G and finally by a call to MXV the transformed intensity vectors are calculated. After the entire image has been processed the statistical parameters associated with the principal components are displayed on the TEKTRONIX.

7.4 HIST 2D

HIST 2D is an acronym for 2 dimensional intensity histogram. Formulaation of the PM was based on two objectives. The first was to visually present the distribution of intensity/feature pairs for scenes (many images of the same geographic area) or subscenes. The second was to present the similarity/separation between subscenes using only two features (two elements of the pixel vector) of the scene. Motivation for using only two features was in part based upon analysis results of LANDSAT multispectral scanner (MSS) imagery. There, by means of the Karhunen - Loève transformation or some other appropriate transformation the two most significant principal components usually contain more than 95 percent of the information in the original four bands.

Conceptually the determination of the distribution of intensity pairs, a two dimensional intensity histogram, is very simple. For every pixel of interest one forms a duple based upon which channel (element of the pixel intensity vector) the user selected for the $\tilde{x}$ variable and similiarily for the $\tilde{y}$ variable. This duple identifies a particular counter which is incremented for each occurrence of the particular duple. Software implementation, however, is not so straightforward. For example, if the pixel's intensity value has a dynamic range of 256 (common 8 bit pixel data), then 256 x 256 counters are required to develop the distribution. Another complexity is the COMTAL display of the distribution because the range of the counts (which is based on the size and characteristics of the scene) is from zero to tens of thousands or more. These problems have been solved and this distribution is presented on the COMTAL as a black and white image in which the x-axis is presented horizontally and the y-axis vertically. There is a strict mapping between COMTAL coordinates and the 2 dimensional intensity plane. The number of occurrences is mapped into brightness, the greater the number of occurrences the brighter (whiter) the area in the intensity plane on the COMTAL.

The second objective was satisfied by superimposing on the two-dimensional intensity histograms an ellipse of constant probability for each selectable subscene. For this purpose it was assumed that the subscene can be represented by a two dimensional normal distribution, where the random vector (x,y) is defined by the density function.

$$f(x, y) = \frac{1}{\sqrt{2\pi\sigma_1^2\sigma_2^2 - \sigma_{12}^2}} \exp\left\{ - \frac{\sigma_1^2 \sigma_2^2}{2(\sigma_1^2\sigma_2^2 - \sigma_{12}^2)} \left[\frac{(x-\mu_1)^2}{\sigma_1^2} - 2\sigma_{12}\frac{(x-\mu_1)(y-\mu_2)}{\sigma_1^2\sigma_2^2} + \frac{(y-\mu_2)^2}{\sigma_2^2} \right] \right\}$$

where μ_1, μ_2 are the means, σ_1^2, σ_2^2 are the variances for $\tilde{x}$ and $\tilde{y}$, respectively, and σ_{12} is the covariance.

A property of the density function f(x,y) can be geometrically interpreted by intersection the survace f(x,y) by horizontal planes f(x,y) equal to a constant as shown in Figure 7.4-1. The resulting curves of intersection lines form a family of ellipses, the equation of which is

$$h(x,y) = \frac{1}{(1-\rho^2)} \left[\frac{(x-\mu_1)^2}{\sigma_1^2} - \frac{2\rho(x-\mu_1)(y-\mu_2)}{\sigma_1 \sigma_2} + \frac{(y-\mu_2)^2}{\sigma_2^2} \right] = k^2$$

where ρ is the correlation term $= \sigma_{12}/\sigma_1\sigma_2$.

With the origin of the coordinate system shifted to the common center of the ellipses (μ_1, μ_2), Figure 7.4-2 presents some relations for k=1. Such an ellipse is known as an ellipse of constant probability : the value of such probability depends on the selected value of k.
From the above equation, for the case k=1, and the equation of an ellipse, the semimajor and semiminor may be computed from

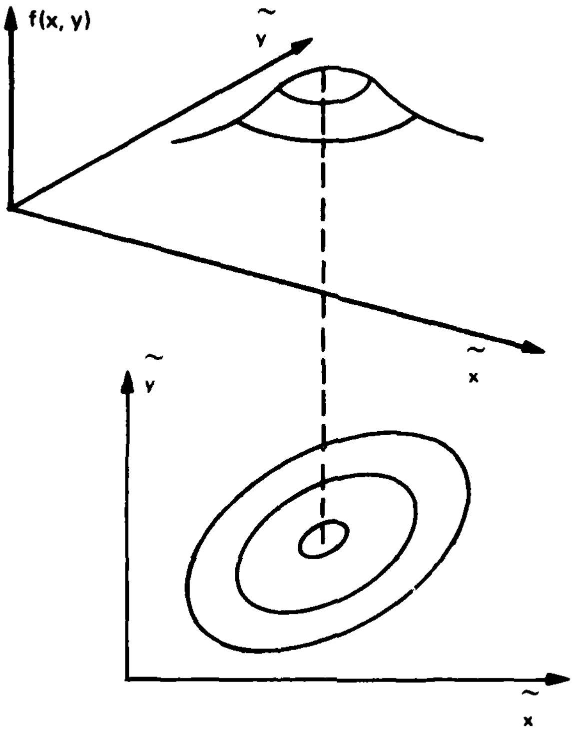
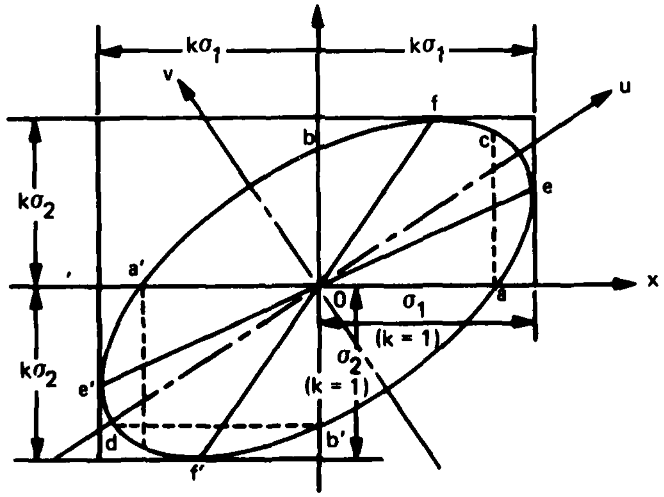


Figure 7.4-1. Properties of Density Function (f(x, y))



$x_a = -x_{a'} = \sigma_1\sqrt{1-\rho^2} = \sigma(x|y)$;
$y_b = y_{b'} = \sigma_2\sqrt{1-\rho^2} = \sigma(y|x)$; $y_c = 2\rho\sigma_2\sqrt{1-\rho^2}$; $x_d = 2\rho\sigma_1\sqrt{1-\rho^2}$; $x_e = x_{e'} = a_1 y_e$
$= -y_{e'} = \rho\sigma_2$; $x_f = x_{f'} = \rho\sigma_1 Y_f = -y_{f'} = \sigma_2$

Figure 7.4-2. The Standard Ellipse. Relations for K=1

$$a^2 = \tfrac{1}{2}\left(\sigma_1^2 + \sigma_2^2\right) + \sqrt{\tfrac{1}{4}\left(\sigma_1^2 - \sigma_2^2\right)^2 + \sigma_{12}^2}$$

$$b^2 = \tfrac{1}{2}\left(\sigma_1^2 + \sigma_2^2\right) - \sqrt{\tfrac{1}{4}\left(\sigma_1^2 - \sigma_2^2\right) + \sigma_{12}^2}$$

The values of a and b can also be obtained from the square roots of the eigenvalues of the covariance matrix,

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{12} & \sigma_2^2 \end{bmatrix}$$

where the characteristic polynominal of Σ is given by

$$\lambda^2 - \left(\sigma_1^2 + \sigma_2^2\right)\lambda + \left(\sigma_1^2\ \sigma_2^2 - \sigma_{12}^2\right) = 0$$

The roots of the above polynominal are those given by the equations for a and b which verifies the relationships between the eigenvalues of Σ and the semimajor and semiminor axes of the standard ellipse.

The angle γ between the semimajor axis of the ellipse and the x axis is given by the following relationship:

$$\tan 2\gamma = \frac{2\sigma_{12}}{\sigma_1^2 - \sigma_2^2}$$

This development has been described in many standard texts, for example Ref. 5 and 6.

The HIST2D processing module displays on the COMTAL for each subscene the ellipse of constant probability corresponding to a probability of 0.90.

It is centered at μ_1, μ_2 and has the orientation as given by the angle γ. Also on the Tektronix the values of μ_1, μ_2, σ^2 and σ^2, the eigenvalues, the eigenvectors and the angle γ are displayed.

7.4.1 User Information

This PM is initiated by entering "HIST2D" on the Tektronix in the same manner as other Dial PM's are started. Once HIST2D has been initiated and the composite image selected, the user is requested to select either the entire image/scene or particular fields/subscenes. Based upon this selection either all pixels in the scene will be used to develop the distribution and ellipse of constant probability or only those in the subscenes. The requests and menus for the entire image case are a subset of those of the subscene case which is described below. Since the scene (composite image) can be multi-dimensional (many channels), the user is requested to select which element of the pixel vector (channel) should be plotted along the x axis and which should be plotted along the y-axis. For the purpose of this PM a subscene is just the union of a set of fields. These fields need to be defined prior to entry of this PM. The field/class file where these fields reside is requested of the user. Next the user is requested to select the fields to make up the subscenes. Actual forming of subscenes is accomplished later in the program so at this time all fields to be used need to be selected. The selection procedure begins with the menu given below.

Select Option

1. Select Field(s) Fields to be Displayed --25Max--

2. Drop Field(s) From the selected list

3. Field Selection completed
(Enter a value from 1 to 3) 1
Where the response should be a "1".

The second menu presented contains a description of the fields, where the required response is the numbers of those fields wanted. When all the fields have been selected and reviewed, the select option menu will appear and a response of "3" will terminate the field selection process.

Once the fields have been selected, the calculation of the two dimensional histogram is started. When this computation is completed the distribution is displayed on the COMTAL as a black and white image, where increasing counts correspond to greater intensity (black to white). Initially a magnification factor of one is used which means that each COMTAL resolution element corresponds to one x,y coordinate of the intensity plane. The center of the COMTAL screen corresponds to the center of the intensity plane. Superimposed on the distribution are ellipses of constant probability (currently set to 0.90), which are determined from the statistics of a union of fields (subscenes). Field selection for a particular contour is from the menu given in Figure 7.4-3. The user need only enter the numbers of the fields that are to make up the contour. In the example, the three oats fields 1, 2 and 3 were selected. After the selection, on the Tektronix is displayed statistics of the union of the fields and on the COMTAL

```
                                   FIELD SELECTION FOR CONTOURS
FLD. NO                FIELD NAME               FLD. NO.               FIELD NAME
      1  OATS247
      2  OATS192
      3  OATS226
      4  CORN211
      5  CORN80
ENTER FIELD NUMBERS  FOR CONTOUR   1    1,2,3

 NO.OF PIXELS =     126  ANGLE BETWEEN EIGENVECTOR AND X AXIS =  56.665

                                     EIGEN
 BAND      MEAN          VARIANCE    VALUES        EIGENVECTORS
-------------------------------------------------------------------------
   1      26.008         2.3919      6.8172       .54954       .83547
   2      41.159         4.9026      .47738       .83547      -.54954
-------------------------------------------------------------------------

 ENTER -X- TO TERMINATE PLOTING, OR -C- TO CONTINUE
```

Figure 7.4-3. Field Selection

the ellipse is displayed with contour identifications. Up to twenty contours can be displayed, and by entering an X the plotting will be terminated. When this happens the user is presented with a set of processing options

Processing Options Are

1. Erase contours displayed and select others

2. List histogram values for an area of interest

3. Magnify an area

4. Terminate processing

Enter number of option

The first option erases the background distribution and ellipses and presents the field selection menu in order to start plotting more ellipses. The second option gives the actual number of intensity duple occurrence for a 20 x 20 range in intensity values centered about a trackball selected point in the intensity display. An example of this display is given in Figure 7.4-4. Magnification of the contours and distribution is accomplished by selecting option 3. The intensity point to be used for the center of the display is selected via the trackball and the magnification factor is entered on the Tektronix. Finally option 4 terminates this PM.

20X20 MATRIX OF HISTOGRAM COUNTS

COMPOSITE IMAGE IS HANDCO

X-AXIS IS CHANNEL 1----------Y-AXIS IS CHANNEL 3

Y-INT.	X-INTENSITY 11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
26---	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
27---	0	0	0	0	0	0	2	0	3	1	0	0	0	0	0	0	0	0	0	0
28---	0	0	0	0	0	1	5	21	11	1	0	0	0	0	0	0	0	0	0	0
29---	0	0	0	0	0	0	1	12	11	9	2	0	0	0	0	0	0	0	0	0
30---	0	0	0	0	0	0	2	2	3	3	2	0	0	0	0	0	0	0	0	0
31---	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
32---	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0
33---	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
34---	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
35---	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
36---	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
37---	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
38---	0	0	0	0	0	0	0	0	0	0	0	0	0	2	1	0	0	0	0	0
39---	0	0	0	0	0	0	0	0	0	0	0	0	0	2	5	9	0	0	0	0
40---	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	7	2	0	0	0
41---	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	10	4	0	0	0
42---	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	2	1	0	0
43---	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	10	6	1	0
44---	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7	13	14	1
45---	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	3	4	2

TO LIST ANOTHER AREA ENTER (CR)

OR TO RETURN TO MAIN MENU ENTER -X-

Figure 7.4-4. Histogram Counts

7.4.2 Control Flow

The control flow of HIST2D is described by the following Program Design Language (PDL).

```
HIST2D:  BGNSEGMENT (MAIN)
CALL LOCATE TO SELECT COMPOSITE IMAGE
CALL TEKVAL TO SELECT PROCESSING OPTIONS
CASENTRY (ENTIRE IMAGE, SELECTED FIELDS)
CASE 1 (ENTIRE IMAGE)
     CALL TEKVAL TO SELECT X AND Y CHANNELS
     DO FROM 1 TO THE NUMBER OF LINES IN COMPOSITE
        CALL DREAD TO READ A LINE
        CALL ACCUMX TO ACCUMULATE COUNTS IN ECS
          FOR THIS LINE
     ENDDO
     CALL DOKNTS TO CALCULATE DISTRIBUTION OF
        COUNTERS
     CALL DSPKNT TO DISPLAY CONTENTS OF COUNTERS
     CALL GLINES TO DISPLAY COORDINATES AXES
     CALL STATH1 TO CALCULATE COVARIANCE MATRIX AND PRINCIPLE
       FACTORS (EIGENVALUES AND EIGENVECTORS)
     CALL EQUCON TO CALCULATE AND DISPLAY ON THE FAST GRAPHICS
       THE ELLIPSE OF CONSTANT PROBABILITY (EQUAL PROBABLE CONTOUR)
     CALL TEKWRT TO DISPLAY SCENE HEADER DATA ON TEKTRONIX
     CALL LSTAT TO DISPLAY SCENE STATISTICS ON TEKTRONIX
     CASENTRY (LIST HISTOGRAM VALUES, MAGNIFY AREA, TERMINATE)
     CASE 1 (LIST HISTOGRAM VALUES)
          CALL LSTAR TO LIST ON THE TEK.  A 20x20 MATRIX OF COUNTS
     END CASE
```

```
          CASE 2 (MAGNIFY AREA)
               CALL MGKNT TO DEVELOP CENTER
                 AND MAGNIFICATION FACTOR
               CALL DOKNTS TO CALCULATE DISTRIBUTION OF COUNTERS
               CALL DSPKNT TO DISPLAY CONTENTS OF COUNTERS BASED ON
                 MAGNIFICATION FACTORS
               CALL GLINES TO DISPLAY COORDINATE AXES
               CALL STATH1 TO CALCULATE STATISTICS
               CALL EQUCON TO CALCULATE AND DISPLAY ELLIPSE
               CALL TEKWRT TO DISPLAY SCENE HEADER DATA
               CALL LSTAT TO DISPLAY SCENE STATISTICS ON TEKTRONIX
          END CASE
          CASE 3 (TERMINATE)
     END CASE
     CASE 2 (SELECTED FIELDS)
          CALL TEKVAL TO SELECT X&Y CHANNELS
          CALL LOCATE TO SELECT FIELD/CASE FILE
          CALL FETEKDF TO SELECT FIELDS
          DO FROM 1 TO NUMBER OF FIELDS
               CALL FEUTLFC TO RETRIEVE FIELD RECORD
               CALL FEUTLLA TO GET MIN AND MAX LINE OF FIELD
               STORE FIELD DEFINITION
          ENDDO
          DO FROM MIN LINE TO MAX LINE OF FIELDS
               CALL DREAD TO READ LINE OF IMAGES
               DO FROM 1 TO NUMBER OF FIELDS
                 IF LINE IS IN FIELD
                 THEN CALL FEUTLLA TO GET PIXEL INTERSECTIONS
               ENDDO
               CALL ACCUMX TO ACCUMLATE COUNTS AND STATS FOR
                 ALL FIELDS INTERSECTING THIS LINE
               ENDDO
```

```
            CALL DOKNTS TO GET DISTRIBUTION OF COUNTERS
            CALL DSPKNT TO DISPLAY CONTENTS OF COUNTERS
            CALL GLINES TO DISPLAY COORDINATE AXES
            DO UNTIL NO MORE CONTOURS
              CALL TEKMSG TO DISPLAY NAMES OF ALL FIELDS
              CALL TEKVAL TO GET FIELD NUMBERS TO MAKE THIS SUBSCENE
                OR CONTOUR
              CALL STATHI TO CALCULATE COVARIANCE MATRIX AND PRINCIPLE
                FACTORS (EIGENVALUES AND EIGENVECTORS)
              CALL EQUCON TO CALCULATE AND DISPLAY ON THE FAST GRAPHICS
                THE ELLIPSE OF CONSTANT PROBABILITY
              CALL TEKWRT TO DISPLAY SUBSCENE HEADER DATA ON TEKTRONIX
              CALL LSTAT TO DISPLAY SUBSCENE STATISTICS ON TEKTRONIX
            ENDDO
            CASE ENTRY (ERASE CONTOURS, LIST HISTOGRAM VALUES, MAGNIFY
              AREA, TERMINATE)
            CASE 1 (ERASE CONTOURS)
              CALL DSPLY TO ERASE IMAGE
              CALL GSET TO ERASE FACT GRAPHICS
            END CASE
            CASE 2, CASE 3, AND CASE 4 ARE THE SAME AS CASE 1, CASE 2,
              AND CASE 3 OF THE ENTIRE IMAGE CASE
        END CASE
    END CASE
    CALL DRETURN TO RETURN COMPOSITE IMAGE
    CALL FINIS TO TERMINATE PM
```

7.4.3 Program Subroutine Description

All subroutines that were developed for HIST2D are described below:

7.4.3.1 Subroutine ACCUMX

Purpose

Accumulate occurrences of intensity duples and field statistics

Usage

```
CALL ACCUMX (BUF, ND, NPIX, IAXIS, IFLDST, JSEGST, JSEGEN, NFLDS,
        NOSEGS, NBIT)
```

Description of Parameters

BUF	-	Input Array with Packed Line of Imagery
ND	-	Input Number of Channels (Vector Dimension)
NPIX	-	Input Number of Pixels in Line
IAXIS	-	Input 2 Word Array First Word X-axis Channel No. Second Word Y-Axis Channel No.
ISEGST	-	Input Array, 12 Words Per Field Gives Starting Pixel for Every Intersection of Line with Field, NOSEGS gives Number of Intersections of Segments. 12 Maximum
ISEGEN	-	Input Array, 12 words per field Same as JSEGST but Having Ending Pixel
NFLDS	-	Input Number of fields (25 Max) per field containing number of segments for this line for this field
NBIT	-	Input Number of Bits per pixel channel
IFLDST	-	Output Array Containing Field Statistics 7 words per Field, All integers Word 1 = Sum of X values Word 2 = Sum of Y values

Word 3 = Number of Pixels in Field

Word 4 = Sum of X*X

Word 5 = Sum of Y*Y

Word 6 = Sum of X*Y

Word 7 = EXTRA

Method

This routine determines if a pixel belongs to one of the selected fields. If it does a duple is formed from the pixel vector. This duple identifies a 12 bit counter in ECS which is read, incremented, and written back to ECS. Storage is allocated for images with 9 bits per pixel or fewer, that is 512 x 512 counters, and for those images exceeding this only the 9 most significant bits are used to identify counters in ECS. Depending upon which field this pixel resides, the statistics from this field are updated to reflect this intensity duple.

7.4.3.2 Subroutine DOKNTS

Purpose

Calculate the cumulative distribution of intensity duple counts

Usage

CALL DOKNTS (IBUF, IHIS, CUMK)

Description of Parameters

IBUF - Input Array supplied by User 512 words in Length to Hold one Row of Counters

THIS - Input Array Supplied by User 256 words in Length to Hold Histogram of Counts

CUMK - Output Array Cumulative Distribution of Counts

Method

In ECS there are 512 x 512 twelve bit counters. The contents of each counter is shifted to the right by 4 bits (dividing by 16) and used as an index into a histogram array. The contents of the histogram array corresponding to the index is incremented by one. From the 256 value histogram a cumulative distribution is calculated.

7.4.3.3 Subroutine DSPKNT

Purpose

Display 2 Dimensional Histogram

Usage

CALL DSPKNT (BUFA, CUMK, THRES, NAMEC, LN, MAGPAR, NBIT)

Description of Parameters

BUFA - Buffer Used for Unpacked Row of Counters Supplied by User 512 words

BUFB - Buffer Used for Packed Row of Counters
Supplied by User 512 words Long

CUMK - Cumulative Distribution of Counter Values - Input

THRES - Threshold for the Number of Counters that Can
Saturate Display - Input

NAMEC - Array Having Name of Composite Image (Input)

LN - Display Name of Histogram (Output)

MAGPAR - Magnification Parameters
MAGPAR (1) = 0 No Magnification
MAGPAR (1) = 1 Magnify
MAGPAR (2) = X Intensity Center
MAGPAR (3) = Y Intensity Center
MAGPAR (4) = Magnification Factor

NBIT - Number of Bits per Pixel

Method

ECS has been loaded with a 512 x 512, 12 bit image, which represents the number of occurrences of particular duples. The cumulative distribution of counters, based upon the 8 most significant bits of the image, and the threshold (THRES) are used to select which 8 bits are to be displayed. One minus the threshold gives the upper limits for the number of duple counters that can saturate the COMTAL. Each line is read from ECS, unpacked, shifted to select bits, repacked and written to the COMTAL via IMGLINE. When magnification is required pixel replication is used for magnifying about the selected intensity center.

7.4.3.4 Subroutine EQUCON

Purpose

Plots on the COMTAL Fast Graphics the Ellipse of Constant Probability

Usage

```
CALL EQUCON (IBUFX, IBUFY, BUFX, BUFY, SM, COV, LN, NBIT, FLDNM,
             DISF, MAGPAR)
```

Description of Parameters

- IBUFX - Integer Buffer (101 words) User Supplied for X-Axis
- IBUFY - Integer Buffer (101 words) User Supplied for Y-Axis
- BUFX - Buffer (101 words) User Supplied for X-Axis Contour Pts, Intensity Values
- BUFY - Same as BUFX but for Y-Axis contour
- SM - Array of Sample Means
- COV - Two Dimension Array of Covariance Matrix
- LN - Display Name
- NBIT - Number of Bits per Pixel
- FLDNM - Field Name 5 Characters
- DISF - CHI**2 Value Corresponding to Per Cent of Distribution Enclosed by Contour
- MAGPAR - Magnification Parameters
  - MAGPAR (1) = 0 No Magnification
  - MAGPAR (1) = 1 Magnify
  - MAGPAR (2) = X Intensity Center
  - MAGPAR (3) = Y Intensity Center
  - MAGPAR (4) = Magnification Factor

Method

Using the equations developed in section 7.4 this routine calculates and plots on the fast graphics the ellipse corresponding to a probability of 0.9. The ellipse coordinates are a function of the sample mean, covariance matrix and the k^2 (DISF) parameter which are all variable. The transformation into COMTAL coordinates takes into consideration the magnification factors and the range of intensity values given by NBIT.

7.4.3.5 Subroutine KBUMP

Purpose

Increments an ECS resident counter based upon an intensity duple.

Usage

CALL KBUMP (IC, IR)

Description of Parameters

IR - Intensity value corresponding to a row number

IC - Intensity value corresponding to a column number (range is 0 to 511)

Method

Based upon the variables IR and IC a particular counter (12 bits in length) in ECS is read and incremented and written back to ECS.

7.4.3.6 Subroutine LSTAR

Purpose

Displays on Tektronix a 20x20 Matrix of Histogram Counts

Usage

```
CALL LSTAR (LN, IAXIS, NAMEC, BUFA, MAGPAR, NBIT)
```

Description of Parameters

LN - Input Image Display Name

IAXIS - Input Array 2 Words

IAXIS (1) = X Axis Channel Number

IAXIS (2) = Y Axis Channel Number

NAMEC - Input 4 Word Array Name of Composite Image

BUFA - Input Buffer of 512 words Used for Packed Row of Counters

NBIT - Number of Bits per Value

MAGPAR - Magnification Parameters

MAGPAR (1) = 0 No Magnification

MAGPAR (1) = 1 Magnify

MAGPAR (2) = X Intensity Center

MAGPAR (3) = Y Intensity Center

MAGPAR (4) = Magnification Factor

Method

The user is requested to position the cursor at the center of the area in the intensity plane for which counts are to be displayed.

These 12 bit histogram counts are retrieved from ECS based upon the cursor coordinate and the magnification parameter, and displayed in the Tektronix.

7.4.3.7 Subroutine LSTAT

Purpose

Tektronix Display of Statistical Parameters

Usage

CALL LSTAT (SM, COV, EIGVEC, EIGPAR)

Description of Parameters

SM - Array of Sample Means (16 Max.)
COV - Covariance Matrix (16x16)
EIGVEC - Eigenvectors Stored Column-wise
EIGPAR - Ordered Eigenvalues

7.4.3.8 Subroutine MAGKNT

Purpose

Determines Magnification Parameters (Center Position and Mag Factor)

Usage

CALL MAGKNT (LN, MAGPAR, NBIT)

Description of Parameters

LN - Input Image Display Name

NBIT - Number of Bits per Pixel Intensity Values

MAGPAR - 4 Word Array - Upon Entry Contains Old Magnification Parameters. All Zero if No Prior Magnification. Upon Leaving Array Contains New Magnification Parameters

MAGPAR - Magnification Parameters

MAGPAR (1) = 0 No Magnification

MAGPAR (1) = Magnify

MAGPAR (2) = X Intensity Center

MAGPAR (3) = Y Intensity Center

MAGPAR (4) = Magnification Factor

Method

Requests the user to position the cursor over the location in the intensity plane that will be the center for the expanded display of the histogram counts and ellipse. Also a magnification factor is requested. Prior magnification is taken into consideration when determining the new intensity center. The intensity center is then correlated with the center of the COMTAL screen.

7.4.3.9 Subroutine PIXEXT

Purpose

Extract One Pixel of ND Values from Packed Line

Usage

CALL PIXEXT (BUF, J, ND, NBIT, IPIXS)

Description of Parameters

- BUF - Integer Array Having Packed Line of Imagery (Input)
- J - Pixel Number (Input)
- ND - Dimensionality of Data (No. of Channels) (Input)
- NBIT - Number of Bits per Value (Input)
- IPIXS - Array Having Extracted Pixel Values one Value per 60 Bit Word

7.5 PLABEL PROCESSING MODULE

PLABEL provides the initial label/class probabilities that are used with the relaxation labeling techniques (See Section 4) for reducing local classification ambiguities. These class probabilities are based upon the observation vector x having a normal distribution and a Bayes rule, which can be expressed by

$$P_i = \frac{p_i}{|\Sigma_i|} \exp \left[-\tfrac{1}{2} (x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i) \right] \tag{1}$$

where p_i is the apriori probability of observing the i^{th} class, μ_i is the mean of the i^{th} class and Σ_i is the covariance matrix of the i^{th} class. For each class a P_i is determined, but instead of storing all of the P's for each pixel, when many will be zero, only a user selectable subset of the most probable are retained. This subset is then ordered, its sum normalized to 1, and stored in a composite image.

7.5.1 User Information

User requests and menus for PLABEL are patterned after those of MAXLIK. For a detailed description of them See Section 3.5.1 of Reference 1. Also the processing steps and their order are very similar for the two PM's. But there are some differences that will be described below.

PLABEL classifies the entire composite image, while MAXLIK classifies only those pixels within a rectangle enclosing the selected fields. Since PLABEL does not have any results presentation capability it does not use or need field descriptions. There is one request that PLABEL makes that is not in MAXLIK. It is for the number of class probabilities to be retained in the composite image for further processing. During class selection the maximum number of classes (MXLABS) was established. This request is asking for how many (NLABS) are to be retained with every pixel (NLABS $\leq$ MXLABS). PLABEL then stores in the composite image NLABS probabilities in descending order of magnitude along with their corresponding label ID. This is in contrast to MAXLIK which only stores the most probable class ID and its chi-squared value (bilinear form). No results other than processing status messages are provided for in this PM, and in order to review the results the user must invoke the ITRES PM.

A probability composite image is the primary output of PLABEL. It is NLABS dimensional, that is there are NLABS 15 bit quantities stored for each pixel. Each record represents a line of the original imagery and contains NLABS times the number of pixels per line 15 bit quantities. The first 5 bits represent the class ID and the next 10 bits contains the corresponding probability times 1023. Label information is as per the DIAL standard with the following additions.

1) Word 4 contains the total number of 15 bit values

2) Word 365 contains NLABS the number labels/probabilities retained

3) There is a photogrammetric parameter record (PPR)

4) First word of PPR, in floating point format, is the maximum number of classes (MXLABS)

5) The next 44 words contain class/label names

6) The next 11 words contain the class character 6 bits left justified, one per word

This is the only vehicle for passing results from PLABEL to RELAX or ITRES. It should be noted that the output image must be saved if it is to be retained after the user logs off.

7.5.2 PLABEL Control Flow

The control flow of PLABEL is described by the following PDL

PLABEL: BGNSEGMENT (MAIN)
CALL LOCATE To Select Classes
CALL FETEKDF To Select Classes
CALL CLASDA To Request Parameters For Classes

```
CALL TEKRD To Select MLC KERNEL
     Case 1  Covariance Matrix
     Case 2  Only Diagonal Elements
     Case 3  Expected Value of Trace
     Case 4  Identity Matrix
CALL LOCATE To Select Composite Image To Be Classified
CALL TEKRD To Determine whether To Display a Band of The Composite
     Case 1  No Continue
     Case 2  Yes CALL DSPCLA To Display Selected Band
CALL TEKVAL To Select Number of Class Probabilities To be Retained
Prepare Label Data for Output Probability Composite Image
CALL RESMAP To Create Output Image
CALL SIGPREP To Calculate Class Signatures
DO UNTIL ALL Image Lines Have Been Classified
     DO UNTIL ECS is Loaded
     CALL DREAD to Read Line from DISK
     CALL WRITEC To Write Line to ECS
     ENDDO
     DO UNTIL ALL Lines in ECS Have Been Classified
     CALL READEC To Read Line From ECS
     CALL MLCR To Determine Probabilities
     CALL DWRITE To Write Results Line to Disk
     ENDDO
ENDDO
CALL DCLOSE To Close Output Image
CALL DRETURN To Return Composite Image
CALL DRETURN To Return Field/Class Data Set
CALL FINIS To Terminate PM
```

7.5.3 Program Subroutine Description

All subroutines developed for PLABEL are described below.

7.5.3.1 Subroutine LABPAC

Purpose

Store label and probability data into an array

Usage

```
CALL LABPAC (IBUF, JPIX, NLABS, PROBS, LABEL)
```

Description of Parameters

- IBUF - An array used to store a packed line of probabilities and labels
- JPIX - Pixel number associated with probabilities and labels stored in PROBS and LABEL respectively
- NLABS - Number of labels/probabilities stored for each pixel
- PROBS - A floating point array containing the probabilities for pixel JPIX
- LABEL - A integer array containing the labels for pixel JPIX

Method

This routine is called once per pixel. It determines where within IBUF, based upon 15 bit values, pixel number, and NLABS, to store the probabilities and labels. The probabilities are scaled by 1023 and the ID is placed in bits 10 to 14. This 15 bit quantity is then placed in the location determined above. This process allows the output line/record to be developed in a packed format and therefore to conserve memory.

7.5.3.2 Subroutine MLCR

Purpose

Evaluates equation 1 for each class and stores a subset of the most probable classes in the output array, which corresponds to a line of imagery.

Usage

```
CALL MLCR (BUFIN, BUFQUT, NLABS)
```

Description

BUFIN - User Supplied Buffer with Packed Line of Imagery
BUFOUT - Output packed line of results (probs and Labels)
NLABS - Number of Labels to Store in Results Image

Remarks

The upper triangular part of the class covariance matrix is stored row-wise in N(N+1)/2 successive storage locations, with the next class matrix starting in the next location of ACOV. Mean vectors are also stored consecutively. Storage has been allocated for up to 10 classes and 24 bands/channels. Pixel intensity values are assumed to be in the band interleaved format.

Method

The likelihood functions is calculated for each class. Based upon the value of KOPTP different approximations to the covariance matrix will be used.

KOPTP - Processing Option

= Use Covariance Matrix

= Use Diagonal Elements of Covariance Matrix

= Use Expected Value of Trace of Covariance Matrix Times Identity Matrix

= Use Identity Matrix

The values of the likelihood function are then ordered from the smallest to the largest. The first NLABS of these are then exponentiated, normalized and stored in the output buffer along with their ID's.

7.5.3.3 Subroutine RESMAP

Purpose

This subroutine creates the output probability composite image and stores appropriate data in the label.

Usage

```
CALL RESMAP (NLABS, IFLAG, PHOTO)
```

Description

NLABS - Number of probabilities/labels to be stored with each pixel in the output image

IFLAG - Flag indicating if this output image is being newly created or if it is an update/iteration of another image
= 0 Newly Created
= 1 Iteration Update

PHOTO - Floating point array with data to be stored in the photogrammetric record.

Method

Uses the standard DIAL routines to create an image with a photogrammetric record. Word 365 is NLABS, and the word that contains the number of pixels now has the number of pixels times NLABS.

When IFLAG=1 the function memory and pseudocolor table are obtained from the prior image.

7.6 RELAX PROCESSING MODULE

RELAX (Relaxation) provides the means for updating the probability estimates for a pixel's labels/classes. These probabilities interact through a set of compatibility functions defined over neighboring pixels. Section 4 describes the interative algorithm used by RELAX and gives references on relaxation labeling techniques.

Software developed for this relaxation activity had the objective of providing a framework for not only the algorithms given in Section 4 but also those suggested by various analysis tasks and a literature search. That is what precipitated the idea of three PM's, PLABEL, RELAX, and ITRES.

- PLABEL develops the initial probabilities and probability image
- RELAX updates the probabilities and creates a new probability image
- ITRES presents summary results of either the initial probabilities or the updated probabilities.

Therefore this section will not only describe how to use this PM but will also highlight areas of the code where new algorithms can be inserted and how this is to be accomplished.

7.6.1 User Information

RELAX starts with a request for a probability composite image. The source if this image could be PLABEL, a prior RELAX run, on some other PM as long as it is in the format described in Section 7.5. RELAX performs one iteration and requires a name for the output results probability image, which is the next request of the user. Following this a request is made for the number of border pixels to be used by the algorithm. Entering a one here indicates that pixels once removed from the pixel being updated influence the update, or an 8 pixel window. If a two was entered a 24 pixel window would be used. This ends the mandatory requests and the remaining are algorithm specific. For the Rosenfeld algorithm the following requests are made:

- ALPHA is the covergence speed-up factor and is usually an integer between 1 and 10. If this is a iteration of prior RELAX results a -CR- will cause the prior value to be used.

- Compatability Values. These are floating point values between -1.0 and 1.0. If this is an iteration of prior RELAX results entering a -Y- will cause the prior values to be used. If this is the first iteration or an -N- was entered for the above, the user is asked if he wants to use a default identity matrix. For the default case the compatibility between the class and itself is one while the compatibility between two different classes is zero. As an example:

	Class A	Class B	Class C
Class A	1	0	0
Class B	0	1	0
Class C	0	0	1

If the default case is not used then the user supplies the compatibility values. Using the above example three requests would be made.

<u>Request 1</u>

Requests the user to supply the compatibility of

    Class A with the Class A
    Class A with Class B
    Class A with Class C

Class A

    Class A
    Class B
    Class C

Enter 3 floating point values between -1 and 1.

Request 2

Class B

    Class B

    Class C

Enter 2 floating point values between -1 and 1.

Request 3

Class C

    Class C

Enter 1 floating point value between -1 and 1.

The number of requests is based upon the number of labels/classes, and the compatibility matrix is assumed to be symmetric.

- Distance Weighting Values. These are floating point values between 0 and 1 describing the importance of a particular pixel neighbor in updating the center pixel. Again a -CR- will cause the prior values to be used for this update. The default case is one over the Euclidean distance between the center pixel and the neighbor. An example for the case when there are two border pixels is given in Table 7.6-1. Also if the user wants to supply his own values he can. For the case given in Table 7.6-1 the user would be requested to make five entries on each of five requests. Each request would be for a particular x column starting from -2 to +2.

If another algorithm was inserted into this PM the above requests might not be necessary, and could be replaced by requests for the parameters of the new algorithm. Output of this PM is the updated probability map and a final processing summary displayed on the Tektronix. The information is basically self explanatory with the

Y \ X	-2	-1	0	1	2
-2	.35	.45	.5	.45	.35
-1	.45	.71	1.0	.71	.45
0	.50	1.0	0	1.0	.5
1	.45	.71	1.0	.71	.45
2	.35	.45	.5	.45	.35

Table 7.6-1. Default Distance Weighting for Two Border Pixels

exception of the definition of interior pixels. This PM does not update the probabilities of a border around the image, the size of which is based upon the number of border pixels. Interior pixels are those that have their probabilities updated. It should be noted that this PM is computationally intensive, and when it is entered at a terminal, batch jobs on the 6400 have to wait until RELAX is completed.

7.6.2 RELAX Control Flow

The control flow of RELAX is described by the following PDL:

```
RELAX:  BGNSEGMENT
CALL LOCATE to select probability composite image
Extract parameters from photogrammetric record
CALL TEKMSG to select number of border pixels
CALL TEKMSG to select ALPHA
CALL TEKMSG to get compatibility values
CALL TEKMSG to get distance weighting matrix
Insert algorithm parameters into photogrammetric record.  See Table
7.6-2 for contents of record.
CALL RSMAP to create output probability image
Load ECS with First Set of Input Prob. Lines
Load Memory with Set of Prob Lines from ECS
    Initialize Lines Circular buffer
DO For All Probability Lines
    Update Lines Circular buffer
    CALL READEC to get line from ECS
    Initialize circular buffer for labels and probabilities
    Store First Set of Labels and probabilities in common arrays
```

```
DO for All Pixels in a Line
   *CALL RSNFLD to update probabilities
    CALL LABPAC to pack updated probabilities and labels in output line
    Gather statistics on updated data
    Update circular buffer for labels and probabilities
    Store next column of labels and probabilities in common arrays

ENDDO
    CALL WRITEC to write line to ECS
IF lines remaining in ECS process next line
IF no lines remaining in ECS unload ECS to disk
IF lines remaining to be processed load ECS and process next line
IF no lines remain to be processed
       Calculate image statistics
       CALL TEKMSG to display stats
       CALL DCLOSE to close output image
       CALL DRETURN to return input probability image
       CALL DRETURN to return input Field/Class file
       CALL FINIS to terminate PM
ENDIF
```

7.6.3 Program Subroutine Description

The subroutines developed for RELAX are described below.

*If a new algorithm is to be used it would simply replace RSNFLD

Table 7.6-2. Contents of Photogrammetric Record

General Description	Starting Word No.	No. of Words	Type
1. Maximum No. of Classes	1	1	F.P.
2. Class Names 4 words per Class	2	44	A/N
3. Class Characters 1 word per class	45	11	A/N
4. Parameter Flags Flag = 0 parameter not in label Flag = 1 parameter in label Flag (1) = Border Flag (2) = ALPHA Flag (3) = Compatability Matrix Flag (4) = Distance Weighting	55	4	F.P.
5. Number of Border Pixels	100	1	F.P.
6. ALPHA	101	1	F.P.
7. Compatability Matrix 10 X 10 values stored by column	102	100	F.P
8. Distance Weighting Matrix No. of Border Pixels X No. of Border Pixels Stored by Column	201	Variable	F.P.

7.6.3.1 Subroutine UNPKPL

Purpose

Unpack and Normalize Pixel Probabilities and Labels

Usage

CALL UNPKPL (PKPL, JPIX, NLABS, PROBP, LABELP)

Description of Parameters

PKPL - An Input Array Containing One Line of a Packed Labels and Probabilities (NLABS Per Pixel)

JPIX - Input Variable containing Pixel Number

NLABS - Input Variable describing Number of Labels/Probs Stored per Pixel

PROBP - Output Array Containing Probabilities

LABELP - Output Array Containing Labels Corresponding to Probabilities

Method

For each pixel there are (NLABS) labels and probs stored in the packed line. One label and prob. take 15 bits, first 5 bits for label and next 10 bits for prob. For the particular pixel the labels and probabilities (normalized) are stored in the output arrays, one word per label and probability.

7.6.3.2 Subroutine RSNFLD

Purpose - applies the nonlinear stochastic relaxation algorithm to a given pixel assumed to be at the center of a square array of pixels.

Usage

CALL RSNFLD (C,R, ALPHA, NEWPRB, NEWLBL)

Description of Parameters

Input

C - IBORDT * IBORDT array of unnormalized pixel weighting coefficients

R - MXLABS * MXLABS array of label (class) compatibilities (all entries between -1. and +1.)

ALPHA - covergence acceleration exponent (default =1)

Output

NEWPRB - NLABS-array of (normalized) updated probabilities

NEWLBL - NLABS-array of corresponding labels (classes)

Method

The probabilities and associated labels are updated according to the nonlinear stochastic model of Rosenfeld et al. as described in Ref. 2, Section V, or Ref. 3, Section III.

7.7 ITRES PROCESSING MODULE

ITRES, which stands for iteration results, displays on the COMTAL and Tektronix summary results of the processing performed by PLABEL and RELAX. It is option driven and uses the prior generated probability composite image for calculating the presented results.

7.7.1 User Information

After ITRES has been initiated the user is requested to enter the name of the probability composite image from which all of the results are obtained. All processing from this point on will be controlled by the option selected from the following list:

1. Field Results

2. COMTAL Display of the Probability Image

3. Tektronix Display of Most Probable Class Assignments for a Pixel Window

4. Terminate PM

- Field Results - The purpose of this option is to gather information on the total image and various fields within the image. Figure 7.7-1 is an example of the information gathered for the total image. The first line gives the name of the results image; in this case it is RELAX1STITERATION. The next line gives the following:

  - The average probability of the most probable class (.918039).

  - The standard deviation associated with the average probability (.132753).

- The average entropy <H>

$$\langle H \rangle = -\frac{1}{K}\sum_{j=1}^{K}\sum_{i=1}^{NLABS} p_i \log p_i$$

  where K is the number of pixels in the image

- The standard deviation associated with the average entropy (.25243).

- The number of pixels in the composite (80000).

The next six lines describe for each class the number of pixels for which this class was most probable and what percent of the total image this comprised.

ITRES

Iteration Results

Results for Image RELAXISTITERATION

PROB = .918039 .132753 Entropy .196352 .252430 Pixels = 80000

Class Name, Number in Class and Percent of Field

1	CORN/4	3900	4.8750
2	SPWH/4	6549	8.1863
3	OATS/4	913	1.1413
4	GRPS501/4	44509	55.6363
5	GRPS504/4	20003	25.0038
6	GRPS505/4	4126	5.1575

To Advance to Next Page of Field Results

Enter ·CR·

Figure 7.7-1. Total Image Results

```
                        RESULTS  FOR IMAGE RELAX1STITERATION

FIELD NAME -------  CORN2
 PROB = .999231 .002772 ENTROPY   .424856E-02 .153184E-01  PIXELS =        14
                          CLASS NAME, NUMBER IN CLASS AND PERCENT OF FIELD
  1  CORN/4                                 14  100.0000
  2  SPWH/4                                  0    0.0000
  3  OATS/4                                  0    0.0000
  4  GRPS501/4                               0    0.0000
  5  GRPS504/4                               0    0.0000
  6  GRPS505/4                               0    0.0000

FIELD NAME -------  CORN1
 PROB =1.0000000.000000 ENTROPY  0.          0.            PIXELS =        63
                          CLASS NAME, NUMBER IN CLASS AND PERCENT OF FIELD
  1  CORN/4                                 63  100.0000
  2  SPWH/4                                  0    0.0000
  3  OATS/4                                  0    0.0000
  4  GRPS501/4                               0    0.0000
  5  GRPS504/4                               0    0.0000
  6  GRPS505/4                               0    0.0000

FIELD NAME -------  CORN3
 PROB = .983095 .067985 ENTROPY   .406228E-01 .127556E+00  PIXELS =        83
                          CLASS NAME, NUMBER IN CLASS AND PERCENT OF FIELD
  1  CORN/4                                 81   97.5904
  2  SPWH/4                                  0    0.0000
  3  OATS/4                                  0    0.0000
  4  GRPS501/4                               0    0.0000
  5  GRPS504/4                               2    2.4096
  6  GRPS505/4                               0    0.0000

                          TO ADVANCE TO NEXT PAGE OF FIELD RESULTS ENTER -CR-
```

Figure 7.7-2. Field Results

LABEL/PROB MAP RESULTS
COMPOSITE IMAGE ---- LISH
PIX MIN 1 PIX MAX 308 LINE MIN 1 LINE MAX 25

```
LINE                                 PIXEL
          1         11        21        31        41        51        61        71        81        91
          I---------I---------I---------I---------I---------I---------I---------I---------I---------I---------
  51---DDDEDDDDDDDDDDDDBDDDDDDDDDDDDEEDDDDDDDDBBFDDDDDDDDDDDDBDDDDDDEEDDDEEDDDDDDDDDDDDDDDDDDDDDDDDDDEDDDEEEDDED
  52---FDDDDDDDDDDDDDDDDDDDDDDDEEDDDDDDDDDDDDDDFDDDDDEDDFBBFFFBDDDDDEEEDDEEDDDDDDDDDDDDDDDEDDDEDDDDDDDDDDDDDDDD
  53---DDDEDDDEDDDDDDDDDDDDDEEDEEEDDDDDDDDDDDDDDEDDFFBBFFFFBBDDDDDDDEEEDEEEEDDDDDDDDDDDDDDDEDDDDDDDDEEDDDDDDDDDDD
  54---FDEEDEEEDDDDDDDDDDDDEDDDEEDDDDDDDDDDDDDDEEDDDFFFFBDDDDDDDDDDDDDEEDDEEEDDDDDDDDDEEEEEDDDDEEDDEEDDDDDDDDDDD
  55---FDEEDEEDDDDDDDDDDDDDDDDDDDDDDBDBDDDDEDDDDDDDDDDDDDDBDDDDDDDDEDDEEEDDDDDEDDEEEEDDDEEEDDDDDEDDDDDDDDDD
  56---BDEEEDEEDDDEDDDDDDDDDDDDDDDDDDDFFFFDDDDEEDEDDDDBBFFBBFBDDDDDEEEDEEDDDDDDEDDDDDDDDDDDDDDDDDDDDDDDDDDD
  57---BDEEEEEEEEEEDDDDDDDFFBDDBDDDDDDBFFDDDDDEDDDDBFFBBBFBBFBDDDDEEEEEEDDDEDEDDDEDDDDDDDDDDDDDDDDDDDDDDDDDD
  58---FDDEEEEEEEEEDDDDDBBFFFBBBBDDDDBDDDDDDDDDDBBBBBBBBBFBDDDDDDDDDEDDDDDEEDEDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
  59---FFDEEEDDDDEEEEEDBBBBBBBFFBDDDDDDEDDDDFBBBBBBBBDDDEEEAEDDDDDDDDDDDEEEEDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
  60---BFDEDDDDDDEEEEEDDDDFBBBBDDDDDDDDDDDDBFFBBDDDDEEEEEEEEEEDDDDDDDDDDDDEEEEDDDDDDDDDDDDEEDDDDEEDDDDDDDDDDDD
  61---DDEEDFDDEEEEEEEEEDDDEDBBDDDDDDEEDFFFBBBBFDEEEEAEEEEDDDDDDDDDDDDDDDDDDDDDDDDDDDEEDDEEDDEEDEEEEDDDDDDDDDD
  62---DBDEDBDEEEEEEEEEEDDEEDBBDEDDDDBFBBBFBBBFDAAEEEDDDDDBBBBBDDDDDDDDDDDDDEEEDDEEEEEEEEEEDDDEEEEDDDDDDDDDDDD
  63---DBDEDBFDEEEEEEEEEDEEEFBDDFFFFBBBBBBFBBFDEEEDDFBBBBBBBBBBBDDDDDDDDDDEEEEEEEEEEEEDDDEEDDEEEEDDDDDDDDDDDD
  64---DBDEDBFDEEEEEEEEEDEEEFBBFBBBBBBFBDDBFBDDEEEDBBBBBBBBBBBBBBDDDDEDDDDEEEEEEEEEEEEEDEDDEEDDDDDDDDDDDDDDDDD
  65---DBDEDFFDDEEEEEEEEDEEEFBBBBBFFFFDDDEDDDDDDDDBBBBBBBDBBBDDDDEEEEDDEEDEEEEEEEEDDDDEEEEDDDDBDDDDDDDDDDDDD
  66---EDDDEDFDDEEEEEEEEDDEEEDBBFFFFBBBDDDDDDDDDDDBBBBBFBDEEDDEDDDEEEEDDEEDEEEEEEDDDFDDDEEEEDDDDDDDDDDDDDDDDD
  67---EDBDEDBBDEEEEEEEEDDEEEFFFBBBBBBBBDBDDDBFFDDFBBBDDDDDDDDEEEDDDDEDDDDEEEEAAEEDFFFBDDDDDEEDDDDDDDDDDDDDDDD
  68---ADBDEDBBDEEDDFDDFDDDFBBBBBBFDBBBDBBBFFDDDBDDDDDDDDDDDDDDDDEEEEEEEEEEEAAAEEDBFBBBBBBDEEDDDDDDDDDDDDDDDDD
  69---ADBDDDBBBDDBBBBBBBBBBBBBBBBBBBBBBDDDDDDDDDDDDDDDDDDDDDDDDDDEEEEEEEEEEEEEEEEAEBBBBBBBBBDEDDDDDDDDDDDDDDDDD
  70---AEBDDEDFDDDBBBBBBBBBBBBBBBFFFFDDDDDDDDDDDBDDDDDDDDDBDDDDDDDEEEEEEEEEEEEEEEEAAEDBFBBBBBDDDBDDDDDDDDDDFDDDDDD
  71---AEDFDEDFDEDFBBBBBBBBBBBBBBDDDDDEEEEEEEEEDDBDDDDDDDDBBDDDDDDDEEEEEEEEEEEEEEEEAAAEBFBBBDBDDDBDDDDDDDDDDDDDDDD
  72---AADBDEDFDEDBBBBBBBBBBBBBBDDEEEEEEEEEEEEEEEDDDDDDDDDDDDBDDDDDDDEEEEEEEEEEEEAAAAEEEDDBBDDBDDDDDDDDDDDDDDEEDDD
  73---AADFDEDBFFFBBBBBDDDDDDDEEEEEEEEEEEEEEEEEEEEDEEDDDDDDDDDDDDDDDDEEEEEAAEEEEEEEEEEDDDBDDDDEEDDDDDDDDDDDEEEEDDD
  74---AAEDBDDDBBBBBBBBBDDEEEEEEEEEEEEEEEEEEEEEEEEEEDEEEDDDDDDDDDDDDDEEEEAAEEEEEEEEEEEDFFFFDDEEEDDDDDDDEEEEEEEDEED
  75---AAEBBBDDBBBBBBFFDEEEEEEEEEEEEEEEEEEEEEEEEEEEEEDDDDDDDDDDDDEEEEEEEEEEEEEEEEEEEEEEEDFDFDDEEDDDDDDDDDDDEDEDEEEE
  76---EAEDBDEDBBBBBBFBEEEEEEEEEEEEEEEEEEEEEEEEEEEDDEEDDDDDDDDDDEEEDDEEEEEEEEEEEEEEEEEDDDDDDDDDDDDDDDDDDDDDDDEEEE
  77---EEEEEAEDBBBBBBBBDEEEEEEEEEEEEAEEEEEEEEEEEEEEEEDDDEEDDDDDEDDDDDDDEEEEEEEEEEEEEEEEEDDDDDDDDDDDDDDDDDDDEDDEEEE
  78---EEEEEAADBBBBBBFDDEEEEEEEEEEEAEEEEEEEEEEEEEEEEEDDDDDDDDDDDDDDDDDDDEEEEEEEEEEEEDEEEDDDDDDDDDDDDDDDDDDDEDDEEEEEED
  79---EEEEAAADBBBBBBFBDEEEEEEEEEEEEEEEEEEEEEEEEEEEEDDDEEEDDDDDDDDDEDDEEEEEEEEEEEEEDEEEEDDDDDDDDDDDDDDDEEDDEEEEEEE
  80---EEEEAAAEBBBBBBBBDEEEEEEEEEEEAEEEEEEEEEEEEEEEEEDDEEEEDDDEDDDEEDDDEEEEEEEEEEEEEEEEEEEDDDDDDDDDDDDDDDDEEDEEEEEEEE
  81---EEEEAAAEBBCBBBFDEEEEEEEEEEEEEEEEEEEEEEEEEEEEDDDEEDDDDDDDDDDDBFFDEEEEEEEEEEEEEDDDEEDDDDDDDDDDDDDDDDEEEEEEEEEDDEE
  82---EEEEAAAADBBBBFDEEEEEEEDDEEEEEEEEEEEEEEEEEEEEDDDDEEDDDDDBDDDFFFFFEEEEEEEEEEEEEEDDDEDDDDDDDDDDDDEEEEEEEEEDDDDDDE
  83---EEEEEAAADBBBBFDEEEEEEDBDEEEEEEEEEEEEEEDDDDDDDDDDDDDDDDDDDDFFDDDFFFFFDEEEEEEEEEDDEEEDEEDDDDDDDDDDEEEEEEEEEDDDDDDD
  84---EEEEEAAAEBBBBFDEEEEEEDFDEEEEEEEEEDDDDDDDFFFDDEDEDDDFFFBDDFFFFFDEEEDDDDDEEDDDDDDDDDDDDDDEEEEEEEEEDDDDDDDDD
  85---EEEEEAAAEBBBBBDEEEDDBDEEEEEEEEEDDDFFDDDDDBDDDDDEDDDDFFFDEBFFFFFEEDDDDDEEDDDDDDDDEDDDDEEEEDEEEEDDDDDDDDDD
  86---EEEEEAAAEDBBFDDDDDDBDEEEEEEEEEEDDDDDDDDDDDDDDDDDEEDDDFFFDEDFFFFFDDDDEEEEEEEDDDDDDDDDEEDEEEDDEEEDDDDDDDFDDD
  87---EEEEEEAAAEDBBBDDDDDDBDEEDDEEEEDDDEDDDDDEEEDFFDEEDDDDFFDEDFFFFFBDDEEEEEEEEDDDDDDDDEEEEDDDDDEEDDDDDDDDDDD
  88---EEEEEEAAEEDBBDDDDDDDDDEEEEDEEEEEEEEEEEEEEEDFFFDDEDDDDFFDEDBFFFFBDEEEEEEEDEEDDDDDDDEDEEDDDDEEDDDDDDDDDDEE
  89---EEEEEEAAAEDBBDDDDEDDEEEEEEEEEEEEEEEEEEEEEEEEEDFFBBFDEEEEDFFBDEDFFFFFDEEEEEEEDDDDDDDDDDEDDDDDDDDDEDDDDDDDDEEEE
  90---EEEEEEAAAEDBBDEEEDDDEEEEEEEEEEEEEEEEEEEEEEEEEDFFBBBBFDDEEEDFFFDDDFFFFFDEEEEEEEDDDDEEEDDDDDDDDDDDDDEEDDDDDDDEEEE
```

TO LIST ANOTHER MAP AREA ENTER (CR)

OR TO RETURN TO RESULTS MENU ENTER X

Figure 7.7-3. Most Probable Class Map

Figure 7.7-2 presents the same information but for individual fields. To gather his information the user is requested to enter the name of the field/class file that contains the fields associated witl. the probability composite input. The next request is for the fields and this request is the same as used in MAXLIK, CLUSTER, etc.

- COMTAL Display of Probability Image - This option displays the most probable class of each pixel in the pseudocolor assigned to the class. The user is also asked if the pseudocolor/class assignment is to be reviewed and if the answer is yes then the class name is displayed adjacent to the prior assigned pseudocolor.

- Tektronix Display of Most Probable Class - This option gives the user the capability to display on the Tektronix a 100 pixel by 40 line area of most probable class assignments. The area center is determined by having the user position the cursor on the COMTAL display having the probability image over the center of the area of interest and depressing the "select" or "done" button. An example of this display is given in Figure 7.7-3, where each character represents a particular class. This character selection was performed during PLABEL.

7.7.2 ITRES Control Flow

The control flow of ITRES is described by the following PDL.

```
ITRES:  BGNSEGMENT (MAIN)
CALL LOCATE to select probability composite image
DO UNTIL case 4 is selected
CASENTRY (Field results, COMTAL display of probability image,
         Tektronix display of most probable class,
         Terminate PM)
CASE 1 (Field Results)
     CALL LOCATE to select field/class file
     CALL FETEKDF to select fields
     DO for every line in image
     CALL DREAD to read line from disk
     CALL ACUSTS to accumulate statistics for total image
     DO for every field
     CALL ACUSTS to accumulate stats for field
     ENDDO
     ENDDO
     Calculate total image stats
     CALL TEKMSG to display total image stats
     DO for every field
     Calculate field starts
     CALL TEKMSG to display field starts
     ENDDO
CASE 2 (COMTAL display of probability image)
     CALL DSPMAPP to display probability image
     END CASE
CASE 3 (Tektronix display of most probably class assignment)
     CALL PROBTK to display class characteristics
     END CASE
CASE 4 (terminate)
     CALL DRETURN to return data sets to system
     CALL FINIS
```

7.7.3 Program Subroutine Description

The subroutines developed for ITRES are described below:

7.7.3.1 Subroutine ACUSTS

Purpose

Accumulates field statistics

Usage

CALL ACUSTS (BUF, MXLABS, NLABS, JSEGST, JSEGEN, NOSEGS, STSFL)

Description of Parameters

- BUE - Array having packed line of labels and probabilities
- MXLABS - Range of labels - maximum No. of different labels
- NLABS - Actual number of labels stored per pixel, ordered by probability
- JSEGST - Array with starting pixel of field intersection with line
- JSEGEN - Same as JSEGST but has ending pixel No.
- NOSEGS - Number of segments for this line for this field
- STSFL - Array for accumulation of field stats
  - Word 1 thru MXLABS counts per most probable label
  - Word MXLABS +1 number of pixels in field
  - Word MXLABS +2 probability
  - Word MXLABS +3 probability squared
  - MXLABS +4 entropy
  - MSLABS +5 entropy squared

Method

This routine accumulates the frequency that each label/class is the most probable. It also calculates the sum of the most probable probability and the sum of the most probable probability squared. It determines the entropy, the entropy sum, and the sum of the entropy squared.

7.7.3.2 Subroutine DSPMAPP

Purpose

Displays on the COMTAL the most probable class of each pixel in the pseudocolor assigned to the class.

Usage

```
CALL DSPMAPP (NAME, NACLAS, A, NUMCL, NLABS, ILINE)
```

Description of Parameters

NAME	- Input array with probability image name
NACLAS	- Array with class names, each name is 4 words in length
A	- User supplied buffer for label (2602 words)
NUMCL	- Number of classes
NLABS	- Actual number of labels stored per pixel ordered on probability
ILINE	- User supplied buffer for COMTAL line of imagery (512 words)

Method

Most probable class assignments developed by PLABEL or RELAX are displayed using the function memory and the pseudocolor table stored in the label.

7.7.3.3 Subroutine PROBTX

Purpose

Display on Tektronix a 100 pixel by 40 line area of most probable class assignments.

Usage

CALL PROBTX (NAME, CLASCH, ICOR, NLABS, IBOF)

Description of Parameters

NAMEC - Array with name of probability image

CLASCH - Array having the character associated with each class. Character is stored one per word in display code and left justified

ICOR - Probability image definition
- ICOR(1) - lowest pixel number
- ICOR(2) - highest pixel number
- ICOR(3) - lowest line number
- ICOR(4) - highest line number

IBOF - User supplied buffer for label (2602 words)

NLABS - Number of labels/classes stored per pixel and ordered on probability

Method

The center of the area for detailed pixel assignment review is obtained via the cursor. This is converted to image coordinates from which the pixel window is determined. The class assignment is retrieved from disk for this window and is converted to display code via the CLASCH array and displayed on the TEKTRONIX. The user can select as many areas as desired before returning to the main processing menu.

7.8 MAXLIK INTERFACE TO STARAN

7.8.1 User Information

If the user selects the option to perform the Maximum Likelihood Classification (MLC) on the STARAN, a portion of MAXLIK that interfaces with the STARAN will be invoked. From the user's point of view, only a menu selection of STARAN processing is required. However, the user should realize the restrictions on the type of data that the STARAN software for the MLC will accommodate:

1. Each pixel-word size must be 8 bits, and
2. There can only be 4, 8, 12, or 16 channels.

If the STARAN is selected for processing, and the data is different than above, MXLIK will take a program exit.

7.8.2 MAXLIK Control Flow

If Image Data Cannot be Processed by STARAN Software

```
THEN Take MAXLIK Exit
ENDIF
Calculate Values for Job Control and Data Description Block
Prepare STARAN for processing
CALL SIGSTAR to prepare class signatures for the STARAN
Enter all of the calculated values in the Job Control and Data Descrip-
tion Block including Class Constants
CALL SCNTRL to send Job Control and Data Description Block to the STARAN
CALL SWRITE to send class signatures to the STARAN
CALL BLKBLD to build the first block of image data from the STARAN
DO FROM 1 to the Number of STARAN Data Blocks
     CALL SWRITE to send a block of data to the STARAN
     IF NOT the First Block sent
     THEN CALL RESBLD to reconstruct block of STARAN processed data
     for the class map.
     ENDIF
     CALL SREAD to receive a block of processed data from the STARAN
     IF Not the Last Block sent
     THEN CALL BLKBLD to build a block of image data for the STARAN
     ENDIF
ENDDO
CALL RESBLD to reconstruct the last block of STARAN processed data for
the class map.
CALL DETACH to disconnect the STARAN from MAXLIK
```

7.8.3 Program Segment Description

The interface segment begins by requesting the user to select whether the MLC will be processed on the 6400 or the STARAN. If the user selects the STARAN then the following segment is executed. The composite data set parameters are checked to determine whether they are:

1. Pixel-word of 8 bits, and
2. Number of channels of 4, 8, or 16

If not, A MAXLIK program exit is taken.

Following this a series of values are calculated that are required for part of the Control and Data Description Block. The STARAN is prepared for processing by connecting the STARAN to MAXLIK through a CALL ATTACH, and the to be executed STARAN program is loaded. CALL SIGSTAR to prepare class signatures in the required STARAN format. Load previously calculated values for the Control and Data Description Block into the Block. Convert class constants to STARAN format and load in Control and Data Description Block. Send Control and Data Description Block to the STARAN. Send class signature data to the STARAN. Build the first image data block for the STARAN. Then begin the basic processing loop to send data to the STARAN, reconstruct previous STARAN processed data, receive data from the STARAN, and build an image data block for the STARAN. This particular sequence is followed in the processing loop to permit processing in the STARAN and the 6400 to overlap. A CALL IOSTAT is used to synchronize the 6400 and the STARAN again.
See Appendix A for a description of the data interface between MAXLIK (6400) and the STARAN.

7.8.4 Subroutines

7.8.4.1 Subroutine SIGSTAR

Title - Develops class signature block for the STARAN

Parameters

CALL SIGSTAR (ISIGBK, NCHAN, NCLASS, NWDSB)

ISIGBK - Output array with signatures in packed format (returned)

NCHAN - Number of channels in composite image

NCLASS - Number of classes for which signatures are required

NWDSB - Number of 60-bit words in signature block

Description

After class weights are normalized, the major processing loop: a class record is retrieved from the Field/Class file, and then the covariance matrix is inverted and decomposed into two matrices (lower triangular). All elements normalized such that the largest is between 0.5 and 1.0. These are stored in 23 fractional bits. If the element is negative, the 2's complement is stored in the lower 24 bits of the 60 bit word. The means are converted to 16 bit values and stored 2 per 60 bit word (1 bit for sign, 7 bits for integer, and 8 bits for fraction). Means are negatively biased by 128 to permit full use of the sign-integer. After the major processing loop, the data is packed as 32 bit words.

7.8.4.2 Subroutine BLKBLD

Title - Builds a pixel-vector block for the STARAN

Parameters

CALL BLKBLD (NPXBLK, JBLK, PKLINE, PKBLK)

NPXBLK - Number of pixel-vectors in a block

JBLK - Sequence number of block

PKLINE - User supplied buffer to accommodate one packed line of composite imagery

PKBLK - Block of NPXBLK pixel-vectors for STARAN (returned)

Description

Subroutine BLKBLD assembles a pixel-vector block of the next NPXBLK pixels in preparation to be sent to the STARAN.

7.8.4.3 Subroutine RESBLD

Title - Builds a block of a class map image from a block of results generated by the STARAN MLC.

Parameters

CALL RESBLD (NPXBLK, JBLK, PKLINE, PKBLK)

NPXBLK - Number of pixel result vectors in a STARAN block

JBLK - Sequence number of block

PKLINE - User supplied buffer to accommodate one packed line of class data. (16 bits per pixel)

PKBLK - Block of result-pixel-vectors. (32 bits per pixel; 4 bit ID, 20 bit integer and 8 bit fraction for chi-squared value) (returned)

Description

Subroutine RESBLD reassembles the result-pixel-vectors into lines and writes the results to the output disk file.

Section 8

RECOMMENDATIONS

As a result of the development and coding of new algorithms to support multi-channel image classification, and of the opportunity to apply the PMs developed during the first phase of the contract to a significant problem in land cover classification, several areas for further development suggested themselves. These can be divided into two groups, one of which concerns the modification of existing DIAL PMs, the other of which concerns the additional investigation and testing of the new algorithms developed.

8.1 MODIFICATION OF EXISTING PMs

While the application of the collection of classification PMs to the land cover analysis of LANDSAT scenes of the Fort Bliss, TX area was quite successful (see Section 2), the extensive exercising of the PMs revealed certain places in which easily implemented modifications would increase the interactive efficiency. Specifically they are:

- The FIELDEF and CLASSAT PMs should be modified so that the user can exit from the field class file after any page, and not only after the last page of the file. This change would be almost trivial to implement. A desirable but somewhat more complex modification would enable the user to enter the field class file at any page, rather than only the first page, as is the case at present.

- FIELDEF should be modified so that the user has to describe whether or not to SAVE a field as soon as it has been defined. This will prevent the inadvertent loss of fields. CLASSAT already has this feature, which again would be almost a trivial task to add to FIELDEF.

- Add other measure of the distance between classes, in particular add divergence to CLASTAT and give the user the option of selecting any number of classes for the inter-class distance computation, and display the results in matrix form.

- The control structure of MAXLIK should be modified to allow more flexibility in stepping through the PM. At present there is essentially only one path up to results processing. This modification would also allow the user to change the color assignment of the class map without having to re-classify the image, as is the case now. The experience with the Fort Bliss scene showed that class color assignment for optimum results is definitely an iterative process.

- Coding of the computationally intensive part of the MAXLIK computation on the STARAN should be completed. This implementation methodology could also be carried over to the CLUSTER PM and would decrease classification time for the large scenes by a factor of 20:1, which would allow much larger images to be classified interactively than is the case at present.

8.2 FURTHER INVESTIGATIONS

The results of the Rosenfeld relaxation algorithm applied to the classification of a LACIE intensive site were very encouraging.

Therefore it is recommended that this development should be continued as follows:

- The coding of the empirical relaxation algorithm should be completed. The results of applying this algorithm to the LACIE intensive site should be compared with the results obtained with the Rosenfeld algorithm. These results should suggest additional tuning of the empirical algorithm and possible modifications to both algorithms.

- The building of composite images to include non-sensor data such as a elevation or slope data should be investigated. The resultant multi-channel images should be classified and then subjected to the relaxation algorithms. An improved accuracy of classification is expected to result.

- The Fort Bliss experiment should be continued. The whole image should be classified using as a basis the classes already defined. (See Section 2), and the distribution of the classes compared with that obtained in Ref 9. The Fort Bliss scene also would be a prime candidate for the addition of elevation and/or slope data and then using relaxation to improve classifications.

In the literature a new clustering algorithm known as the "Pseudo-F" test has shown great promise on ETL type problems. Implementation of this algorithm on DIAL would only require minor modifications to the cluster PM. This would increase the usefulness of CLUSTER and enable a more accurate unsupervised classification of scenes of interest to be obtained with fewer passes, since the Pseudo-F Test optimizes the number of clusters.

REFERENCES

1. Rice, W.C., Shipman, J.S., and Spieler, R.J. Interactive Digital Image Processing Investigation. U.S. Army Engineer Topographic Laboratory Report ETL-0172, 29 December 1978.

2. Rosenfeld, A., Hummel, R.A., and Zucker, S.W. Scene Labeling by Relaxation Operations. IEEE Transactions on Systems, Man and Cybernetics, Vol. SMC-6, pp 420-433, June 1976.

3. Zucker, S.W., Krishnamurty, E.V., and Haar, R.L. Relaxation Process for Scene Labeling: Convergence, Speed, and Stability. IEEE Transactions on Systems, Man and Cybernetics, Vol. SMC-8, pp 42-48, January 1978.

4. Ahmed, N. and Rau, K.R. Orthogonal Transforms for Digital Signal Processing. Berlin-Heidelberg -New York: Springer-Verlag, 1975.

5. E.M. Mikhail, Observations and Least Squares. IEP-A Dun-Donnelley, New York (1976)

6. D.M. Young and R.T. Gregory, A Survey of Numerical Mathematics Vol II, Addison-Wesley, Reading MA. (1973)

7. Goetz, A.F.H. et al. Application of ERTS Images and Image Processing to Regional Geologic Problems and Geologic Mapping in Northern Arizona. Technical Report 32-1597, Jet Propulsion Laboratory, Pasadena CA. May 15, 1975.

8. Longshaw, T.G., Viljoen, R.P., and Hudson, M.C. Photographic Display of LANDSAT-1 CCT Images for Improved Geological Definition. IEEE Transactions on Geoscience Electronics, Vol. GE-14, pp 66-78, January 1976.

9. Satterwhite, M.B. et al. Photo Analysis of the Relationships Between Vegetation and Terrain Features in South-central New Mexico and Western Texas.

10. Anderson, J.R., Hardy, E.F., Roach, J.T., and Witmer, R.E. A Land Use and Land Cover Classification System for use with Remote Sensor Data. Geol. Surv. Prof. Paper 964, 1976.

APPENDIX A

STARAN/6400 DATA INTERFACE

A.1 INTRODUCTION

This appendix describes the data interfaces between the CDC 6400 and the STARAN when the computationally intensive kernel of the maximum likelihood classification algorithm is evaluated on the STARAN. There are three data interfaces going from the 6400 to the STARAN and one going in the other direction. Each data interface is described below.

A.2 6400 TO STARAN DATA INTERFACES

A.2.1 Job Control and Data Description

This is the first block of data sent to the STARAN and it is sent only once per classification job. It has a length of 50 words and each 60 bit 6400 word is converted into one 32 bit STARAN word via a call to subroutine SCNTRL.

JOB CONTROL AND DATA DESCRIPTION BLOCK

Word #	Data Description
1.	N - Number of Channels (4, 8, 12 or 16)
2.	K - Number of Classes (2 - 10)
3.	Covariance MATRIX Length N x (N+1)/2
4.	Length of All Covariance Matrices K x N x (N+1)/2
5.	Pixel Vector Block Length in Number of 32 bit Words (4080 Max. or 2176 Max Packed 60 bit Words)
6.	Number of Pixel Blocks

Word #		Data Description
7.		Flag - Indicating if Last Block has Non-significant Pixel Vectors Flag = 1 indicates Yes Flag = 0 indicates No
8.		For the Case that Flag = 1. This word has the number of Significant Pixel Vectors in the Last Pixel Block Also Nonsignificant Pixel Vector Values are Set to 255
9.		Number of Mean Pixel Values KxN
10.		Number of significant 32 bit Wds in Signature Block KxN/2 + KxNx(N+1)/2 Used
11.		Spare
12.		Length of Signature Block (2400 - 32 bit Words)
13.		Number of Pixel Vectors in Full Block (960)
14. - 20.		Spare
21. to 30.		Elements of covariance matrix-ℓ- for these elements the number of bits each ℓ has been shifted in order that the maximum ℓ lie in the range 0.5 to 1
31. to 40.	class constants one per class	
41. to 50.		Spare

A.2.2 Signature Data Block

The signature block is also sent only one per classification job, its length is constant but the number of 32 bit words used is given by

$$y = K \times N/2 + K \times N \times (N + 1)/2$$

32 bit

Word	0 — 15	16 — 31
1	MEAN CHAN 1 CLASS 1	MEAN CHAN 2 CLASS 1
⋮	⋮	
N/2	MEAN CHAN N-1 CLASS 1	MEAN CHAN N CLASS 1
	⋮	⋮
KxN/2+1	MEAN CHAN N-1 CLASS K	MEAN CHAN N CLASS K

ℓ_{11} of CLASS 1
ℓ_{21} of CLASS 1
ℓ_{23} of CLASS 1
ℓ_{31} of CLASS 1
ℓ_{32} of CLASS 1
ℓ_{33} of CLASS 1
ℓ_{41} of CLASS 1
⋮
ℓ_{nn} of CLASS 1
ℓ_{11} of CLASS 2
⋮
ℓ_{nn} of CLASS 2
⋮
ℓ_{11} of CLASS K
⋮
ℓ_{nn} of CLASS K

Figure A.2-1. Signature Block Map

The length in terms of 32 bit words is 2400 which is greater than the largest possible y. This block is transferred via a call to subroutine SWRITE using packing mode 3.

The map of the signature block is given in Figure A.2-1 where two channel means are stored in a 32 bit word

0	1 7	8 15	16	17 23	24 31
Sign bit	7 bits Integer	8 bits Fractional	Sign bit	7 bits Integer	8 bits Fractional
MEAN 1			MEAN 2		

and each component of L (ℓ_{ij}) is a 24 bit quantity stored in the 32 bit word

0 7	8	9 31
BLANK	Sign bit	FRACTIONAL 23 bits

A.2.3 Pixel Vector Data

After the STARAN has been initialized and received the first two blocks of data it is ready to start classifying pixel vectors. A pixel vector data block is sent to the STARAN and it returns the classification results for each pixel in the block. This scheme is continued until all pixels in the composite image have been classified. A pixel vector block always contains the same number of pixel vectors (except possibly for the last block) but in terms of 32 bit words its length is a function of the number of channels. Since only 4, 8, 12, and 16 channel data are allowed each pixel requires an even number of 32 bit words (1, 2, 3, or 4). The map for a 12 channel case is given in Figure 2 and this data block is transferred via a call to subroutine SWRITE using packing mode 3.

A.3. STARAN TO 6400

There is only one block format that is sent from the STARAN to the 6400. This is used after the STARAN completes classifying a block of pixels. For each pixel in the block sent to the STARAN, 32 bits are returned. Bits 0 thru 3 contain the class ID and the remaining 28 bits contain the chi squared value. The first 20 bits are integer and the remaining 8 are fractional. This block is transferred via a call to SREAD using packing mode 3.

32 BIT

	Byte 1	Byte 2	Byte 3	Byte 4
Pixel 1	CHAN. 1	CHAN. 2	CHAN. 3	CHAN.4
	CHAN. 5	CHAN. 6	CHAN. 7	CHAN.8
	CHAN. 9	CHAN.10	CHAN.11	CHAN.12
Pixel 2	CHAN. 1	CHAN. 2	CHAN. 3	CHAN.4
	CHAN. 5	CHAN. 6	CHAN. 7	CHAN.8
	CHAN. 9	CHAN.10	CHAN.11	CHAN.12
Pixel Y	CHAN.1			CHAN.4
	CHAN.9			CHAN.12

Figure A.2-2. Pixel Vector Data Map (N=12)
(8 bit byte all integer, Y = 960)

AD-A087 518 IBM FEDERAL SYSTEMS DIV GAITHERSBURG MD F/G 20/6
INTERACTIVE DIGITAL IMAGE PROCESSING INVESTIGATION. PHASE II.(U)
APR 80 W C RICE, J S SHIPMAN, R J SPIELER DAAK70-77-C-0166
UNCLASSIFIED ETL-0221 NL

3 OF 3
AD
A087518

END
DATE
FILMED
9-80
DTIC

Appendix B

FIELD/CLASS FILE

This appendix describes the format and contents of the field/class file which is used to pass field descriptions or class signatures between PMs. It is a DIAL type 5 (Parameter) data set and each record corresponds to either a field or a class.

Fields	Type	Starting Word	Description
FCNAME (4)	A/N	1	Field/Class Name
FCDESC (4)	A/N	5	Field/Class Description
IFCIND	I	9	Field/Class Indicator 0 - Field 1 - Class
IFCGEO	I	10	Field Geo Type 0 - Linear 1 - Areal
NVERT	I	11	Number of Vertices (max. 12)
IVERT(2,13)	1	12	Vertices of fields IVERT (1,K), - Line IVERT (2,K) - Pixel
(spare -16)		38	
INDSTAT	I	54	Statistics Present Indicator 0 - No 1 - Yes
NPIX	I	55	Number of Pixels to Form Class
NCHAN	I	56	Number of channels
CMEAN (24)	R	57	Mean Values
COV(24,24)	R	81	Matrix (triangular) Upper - Covariance Lower - Correlation
NFNM	I	657	Number of fields used to compile this class. 10 max can be recorded
FNMCL(4,10)	A/N	658	Names of fields used to compile this class. 10 max can be recorded
	TOTAL	697	